Curso de Semântica Formal

Marcelo Ferreira

DRAFT
of 19 de outubro de 2018, 15:59

Textbooks in Language Sciences nn

language science press

Textbooks in Language Sciences

Editors: Stefan Müller, Martin Haspelmath
Editorial Board: Claude Hagège, Marianne Mithun, Anatol Stefanowitsch, Foong Ha Yap

In this series:

1. Müller, Stefan. Grammatical theory: From transformational grammar to constraint-based approaches.
2. Schäfer, Roland. Einführung in die grammatische Beschreibung des Deutschen.

ISSN: 2364-6209

Curso de Semântica Formal

Marcelo Ferreira

DRAFT
of 19 de outubro de 2018, 15:59

language
science
press

Marcelo Ferreira. 2018. *Curso de Semântica Formal* (Textbooks in Language Sciences nn). Berlin: Language Science Press.

This title can be downloaded at:
http://langsci-press.org/catalog
© 2018, Marcelo Ferreira
Published under the Creative Commons Attribution 4.0 Licence (CC BY 4.0):
http://creativecommons.org/licenses/by/4.0/
ISBN: no digital ISBN
no hardcover ISBN
no softcover ISBN
ISSN: 2364-6209
no DOI

Cover and concept of design: Ulrike Harbort
Fonts: Linux Libertine, Arimo, DejaVu Sans Mono
Typesetting software: XeLaTeX

Language Science Press
Unter den Linden 6
10099 Berlin, Germany
langsci-press.org

Storage and cataloguing done by FU Berlin

no logo

Language Science Press has no responsibility for the persistence or accuracy of URLs for external or third-party Internet websites referred to in this publication, and does not guarantee that any content on such websites is, or will remain, accurate or appropriate.

Sumário

Prefácio **vii**

Agradecimentos **xi**

1 Significado e condições de verdade **1**
1.1 Significado, verdade e mundo 1
1.2 Relações semânticas entre sentenças 4
1.3 Composicionalidade 6
1.4 Linguagem objeto e metalinguagem 9
1.5 Extensões 10
1.6 Além de extensões e condições de verdade 20
Sugestões de leitura 23
Exercícios 24

2 Predicação **27**
2.1 Funções e a notação lambda 27
2.2 Verbos intransitivos e aplicação funcional 32
2.3 Verbos transitivos 35
2.4 Tipos e domínios semânticos 38
2.5 Verbos bitransitivos 40
2.6 Predicados não verbais 42
2.7 Relações, funções e *currying* 46
Apêndice: mais sobre conversões lambda e alfa 48
Sugestões de leitura 52
Exercícios 53

3 Coordenação e negação **57**
3.1 Coordenação 57
3.2 A flexibilidade da coordenação 61
3.3 Negação 69
Sugestões de leitura 74
Exercícios 74

4 Referência **77**
4.1 Descrições definidas . . . 77
4.1.1 Unicidade . . . 79
4.2 Nomes próprios . . . 82
4.2.1 Nomes próprios como predicados . . . 85
4.3 Pronomes . . . 88
Sugestões de leitura . . . 95
Exercícios . . . 96

5 Modificação nominal **99**
5.1 Modificadores nominais . . . 99
5.2 Modificação adjetival . . . 100
5.3 Mudança de tipos . . . 103
5.4 Conjunção funcional . . . 105
5.5 Modificação e classes de comparação . . . 107
5.6 Orações relativas . . . 109
5.6.1 Vestígios e movimento sintático . . . 115
Sugestões de leitura . . . 119
Exercícios . . . 120

6 Quantificação nominal **121**
6.1 Expressões quantificadoras e indivíduos . . . 121
6.2 Quantificadores generalizados . . . 124
6.3 Determinantes quantificadores . . . 126
6.4 Nomes próprios e descrições definidas como quantificadores generalizados . . . 130
6.5 Outros DPs quantificadores . . . 132
6.6 Coordenação de DPs . . . 136
6.7 Conservatividade: um universal semântico . . . 140
6.8 Quantificação e escopo . . . 147
6.8.1 DPs quantificadores em posição de objeto . . . 147
6.8.2 Interação de escopo entre DPs . . . 148
6.8.3 Negação e escopo . . . 149
6.9 Quantificação e movimento . . . 150
6.9.1 Movimento e escopo . . . 152
6.9.2 Negação e alçamento de quantificadores . . . 158
6.10 Quantificação e mudança de tipos . . . 163
6.10.1 Mudança de tipos e escopo . . . 166
6.10.2 Mudança de tipos, negação e escopo . . . 169

Sugestões de leitura 171
Exercícios 171

7 Ligação **175**
7.1 Pronomes livres e ligados 175
7.2 Correferência vs. ligação 176
7.3 Ligação sem índices nem atribuições 179
7.3.1 Limitando composição e ligação funcional 185
7.3.2 Pronomes livres novamente 186
Sugestões de leitura 188
Exercícios 188

Referências Bibliográficas **191**

Prefácio

O sistema interpretativo que desenvolveremos neste livro parte de três considerações, até certo ponto, intuitivas. A primeira delas é a de que saber o significado de uma sentença (declarativa) parece relacionado a saber as condições para que a sentença seja verdadeira. Se sabemos o significado da sentença "tem mais de dez pessoas dentro desta sala", podemos não saber se ela é verdadeira ou falsa, mas sabemos como o mundo deve ser para que ela seja verdadeira. Sabemos, por exemplo, que é necessário que o número de pessoas no interior da sala seja superior a dez, mas que o numero exato não é relevante. Sabemos, por outro lado, que é irrelevante haver ou não pessoas próximas à sala, ou as pessoas dentro dela serem homens ou mulheres, ou a sala ser grande ou pequena. Enfim, saber o significado de uma sentença se assemelha a saber as condições necessárias e suficientes para que a sentença seja verdadeira, ou, posto de maneira mais concisa, saber suas condições de verdade. Tendo isso em mente, exploraremos a ideia de que as condições de verdade de uma sentença sejam o seu próprio significado.

A segunda consideração é que o significado de uma sentença é produto do significado das palavras que a compõem e da maneira como estas palavras estão agrupadas, ou seja, da estrutura sintática da sentença. Assim, não é surpresa que o significado de "Pedro beijou Maria" difere do significado tanto de "Pedro abraçou Maria" quanto de "Maria beijou Pedro". No primeiro caso, são palavras distintas e, no segundo, são as mesmas palavras arranjadas de maneira distinta. Radicalizando um tanto esta intuição, parece interessante explorar a ideia de que o significado de uma sentença dependa exclusivamente das palavras e da estrutura sintática que compõem a sentença.

A terceira e última consideração é que expressões linguísticas como palavras, sintagmas nominais e verbais, além das próprias sentenças nos remetem a certas entidades ou a certos aspectos do mundo. Uma possibilidade bastante razoável para começar a compreender esse fato é assumir que as expressões linguísticas se relacionam, elas mesmas, direta ou indiretamente, com as coisas ou fatos do mundo, ou seja, elas têm referência. A que exatamente elas referem, é algo a ser decidido caso a caso. Assim, buscaremos um sistema que seja referencial ou, num jargão mais técnico, extensional, mas sem com isso igualarmos significado e re-

ferência (extensão). Ou seja, queremos capturar a intuição de que os sintagmas nominais "o presidente do Brasil em 2006" e "o sucessor do presidente FHC" se referem ao mesmo indivíduo, Lula, mas que nem por isso têm o mesmo significado e, portanto, contribuem de maneiras distintas na derivação das condições de verdade das sentenças em que ocorrem.

Em suma, desenvolveremos um sistema em que o significado de uma sentença equivalha a suas condições de verdade, que seja composicional e extensional. No primeiro capítulo, desenvolveremos um pouco mais essas considerações iniciais. Nos demais, apresentaremos um sistema que atenda a essas considerações. Faremos isso, explicitando como ele é capaz de interpretar uma série de construções e sentenças do português. Antes, porém, algumas breves palavras sobre a concepção e o público alvo deste curso de semântica formal.

Este livro originou-se de uma apostila preparada para um curso de uma semana de introdução à semântica formal, oferecido na Escola de Verão em Linguística Formal (EVELIN), que aconteceu na Universidade Estadual de Campinas em janeiro de 2004. A partir daí, a apostila foi se expandindo e tornou-se a base de um curso semestral de introdução à semântica formal que tem sido oferecido regularmente no programa de pós-graduação do Departamento de Linguística da Universidade de São Paulo. Partes do conteúdo também foram utilizadas em mini-cursos oferecidos na Universidade Federal de Belo Horizonte em 2008, na Universidade Federal do Rio Grande do Sul em 2009, na Universidade Federal do Paraná, como parte do instituto organizado pela Associação Brasileira de Linguística em 2010, e novamente na Universidade Estadual de Campinas, durante a Escola de Inverno em Linguística Formal, em 2013.

As reações (positivas e negativas) dos alunos em todas essas ocasiões foram cruciais no aprimoramento do material e sou extremamente grato a todos eles por essa enorme contribuição. Suas dificuldades, dúvidas e sugestões motivaram inúmeras revisões de forma e conteúdo, que, espero, tenham tornado o texto mais claro e fácil de acompanhar.

Como dito mais acima, o livro tem o intuito de apresentar um sistema semântico composicional, formalizado através de algumas ferramentas lógico-matemáticas. Não se pressupõe experiência prévia com abordagens formais para o significado. Entretanto, espera-se que o leitor já tenha passado por uma introdução geral à linguística e seus níveis de análise (fonologia, morfologia, sintaxe), bem como a uma discussão informal sobre alguns temas centrais sobre o significado (semântica e pragmática). Em sala de aula, poderá ser usado tanto em cursos mais avançados na graduação quanto em cursos de pós-graduação. Fora dela, poderá satisfazer estudantes autodidatas, professores e pesquisadores não apenas de lin-

guística, mas também de áreas afins com interesse na análise e formalização do significado no âmbito das línguas naturais, como filosofia, ciências cognitivas e da computação.

O material empírico coberto nos seis capítulos que seguem o capítulo introdutório (capítulo 1) compreende fenômenos relacionados a predicação (capítulo 2), coordenação e negação (capítulo 3), referência (capítulo 4), modificacão (capítulo 5), quantificação (capítulo 6) e ligação (capítulo 7). Eles devem ser lidos nessa ordem, já que em cada um deles, novas ferramentas analíticas serão apresentadas e, a partir daí, pressupostas e reutilizadas com frequência.

O leitor que tiver assimilado o conteúdo aqui apresentado será capaz de aplicar as técnicas que aprendeu a outros domínios que não serão cobertos, como tempo, aspecto, e modalidade. Será também capaz de compreender boa parte da literatura mais técnica que se encontra nos principais periódicos da área. Ao final de cada capítulo, há sugestões de leitura para os que quiserem aprofundar-se nos temas abordados no texto. Há também exercícios que devem ser feitos logo após a leitura de cada capítulo e que são cruciais para a fixação do conteúdo em questão.

São Paulo, Outubro de 2018

Agradecimentos

Este livro é fruto do que aprendi como estudante e venho aprendendo como professor em minha trajetória acadêmica. Dessa forma, gostaria de agradecer especialmente aos meus professores e alunos, que fizeram dessa dupla experiência em sala de aula uma parte tão importante da minha vida. Dois momentos foram particularmente marcantes. Quando ainda estava em meu primeiro ano de doutorado e dava meus primeiros passos em semântica formal, Irene Heim e Kai von Fintel foram figuras inspiradoras que me mostraram, na prática, o que era um bom curso de semântica formal. Um tempo depois, quando eu ainda dava os primeiros passos como professor, meus alunos de pós-graduação na Universidade de São Paulo foram uma fonte constante de estímulo para refinar minha didática e dar uma cor local a meu próprio material.

Mais recentemente, tive a sorte de ter como leitores de versões preliminares deste livro, Gabriel Othero e dois pareceristas anônimos da Language Science Press, a quem agradeço imensamente. A leitura minuciosa e crítica que fizeram foi essencial para aprimorar a forma e o conteúdo do que está aqui. Agradeço ainda aos editores Stefan Müller e Martin Haspelmath pela receptividade que tiveram quando enviei à editora a proposta de um manual de semântica formal escrito em português.

Por fim, minha família: minha esposa Elaine, meus filhos Henrique e João Pedro e meu pai José Olindo estão sempre proporcionando momentos de alegria e descontração, contraponto perfeito para as abstrações, fórmulas e símbolos que se encontram nas páginas a seguir.

1 Significado e condições de verdade

Semântica é o estudo do significado e Semântica formal é o estudo do significado através do uso de ferramentas e técnicas lógico-matemáticas. Definições assim tão gerais, entretanto, são pouco esclarecedoras e tendem mais a suscitar do que a responder questões (Significado de quê? O que é significado? Que tipo de ferramentas?). O objetivo deste primeiro capítulo é dar uma resposta preliminar a essas questões, explicitando a acepção que o termo *significado* tomará no contexto deste livro e, sobretudo, motivando a relevância de seu estudo através de um método formal. Sendo este um livro de linguística, nosso interesse estará no significado de expressões das línguas naturais, mais especificamente no significado de sentenças e suas partes (sintagmas). Ainda assim, esse é um terreno vasto e, ao final do capítulo, falaremos um pouco sobre alguns tipos de sentenças cujo estudo está além do escopo da teoria que desenvolveremos.

1.1 Significado, verdade e mundo

Um dos fatos mais salientes a respeito das sentenças de uma língua natural é que seus falantes as usam a todo momento para falar sobre o mundo em que vivem. Tal fato parece ligado à intuição de que as sentenças têm significado e que, cientes desse significado, somos capazes de estabelecer conexões entre linguagem e realidade. Por exemplo, imagine que alguém diga a você: *está chovendo em Paris*. Se você souber o significado dessa sentença, e assumindo que a pessoa esteja sendo sincera, você, então, terá aprendido algo sobre o mundo. Da mesma forma, mas no sentido inverso, se você obtiver (digamos, visualmente) informação sobre o tempo chuvoso em Paris, você automaticamente concluirá que a sentença *está chovendo em Paris* é verdadeira. Ou seja, saber o significado de uma sentença nos permite transitar da linguagem para o mundo e do mundo para a linguagem. Note que nada disso seria possível se não soubéssemos o significado da sentença em questão. Isso nos leva a crer que o significado tem um papel crucial na mediação entre linguagem e realidade. Dada a importância desse papel, é justo cobrar de uma teoria semântica que ela lance luz sobre essa relação linguagem-mundo. É sobre esse aspecto da linguagem, chamado às vezes de referencial, que nos

debruçaremos no decorrer deste livro. Nossa premissa é que as expressões de uma língua natural, sentenças em particular, possuem significado, e que a natureza desse significado está na base das conexões que fazemos entre linguagem e realidade. Exploremos um pouco mais esse ponto.

Voltemos à nossa sentença sobre Paris, destacada em (1) a seguir:

(1) Está chovendo em Paris.

Imagine, agora, que alguém diga a você que sabe o significado dessa sentença do português, e que você decida verificar se a pessoa realmente sabe o que ela significa.

Note, em primeiro lugar, que seria injusto perguntar a essa pessoa se a sentença é verdadeira ou falsa. Pode ser que ela não tenha informações suficientes sobre o tempo em Paris neste instante, e que portanto não possa fornecer a você uma informação confiável. Mas isso nada tem a ver com o fato de ela saber ou não o significado de (1). Saber o significado de uma sentença não significa saber se a sentença é verdadeira ou falsa. Imagine agora que nós mostremos várias imagens da cidade de Paris a essa pessoa e perguntemos a ela se a sentença acima seria verdadeira em cada uma das circunstâncias. Agora sim seria justo cobrar dessa pessoa respostas adequadas. Por exemplo, se em uma das imagens não está chovendo, mas nevando, ou se o chão das ruas estiver molhado, mas não estiver caindo água do céu, ou ainda se estiver chovendo próximo a Paris, mas a imagem deixar claro que está fazendo sol na cidade, nosso informante deve dizer que a sentença é falsa. Por outro lado, se a imagem for mesmo de Paris e estiver chovendo, a pessoa deve dizer que a sentença é verdadeira, independente de outros detalhes, como, por exemplo, se as pessoas estão ou não carregando guarda-chuvas, ou se há ou não um arco-íris no céu. Qualquer resposta contrária a essas expectativas serve de indício de que a pessoa não sabe o significado de (1).

Poderíamos capturar essa intuição associando o conhecimento do significado de uma sentença como (1) ao conhecimento das condições para que a sentença seja verdadeira.

Saber o significado de uma sentença é saber as condições necessárias e suficientes para que a sentença seja verdadeira.

Vamos adotar então como hipótese de trabalho a seguinte definição de significado de uma sentença:

O significado de uma sentença é equivalente às condições necessárias e suficientes para que a sentença seja verdadeira.

De forma mais reduzida:

> **O significado de uma sentença é equivalente às suas condições de verdade.**

Será em torno dessa máxima que construiremos nossa teoria semântica. Teorias assim, em que o significado de uma sentença equivale às suas condições de verdade, são chamadas de VERICONDICIONAIS. Nosso objetivo será estabelecer uma maneira de derivar para cada sentença S de uma língua natural um enunciado da seguinte forma:

'*S*' é verdadeira se e somente se *p*

onde *S* é uma sentença (ou mais precisamente, uma estrutura sintática, conforme veremos mais tarde) e *p* descreve condições sobre o mundo.

É importante notar como uma teoria semântica deste tipo é capaz de estabelecer conexões entre as sentenças de uma língua e o mundo em que vivemos. No exemplo acima, vimos que o conhecimento do significado de uma sentença mais o conhecimento das circunstâncias relevantes do mundo leva ao conhecimento da verdade ou falsidade da sentença (também chamado de valor de verdade da sentença). Igualmente, o conhecimento do significado de uma sentença mais o conhecimento de seu valor de verdade leva ao conhecimento sobre algo relativo ao mundo. Dito de outra forma, se sabemos que uma sentença é verdadeira (ou falsa), então sabemos algo sobre o mundo. Esses dois tipos de inferência centradas em nosso conhecimento semântico são imediatamente capturados por uma teoria baseada em condições de verdade. Isso está representado nos esquemas abaixo (abreviamos a expressão *se, e somente se* por *sse*):

'*S*' é verdadeira sse *p*	Conhecimento semântico +
p	Conhecimento de mundo (fatos) =
logo, '*S*' é verdadeira.	Conhecimento do valor de verdade

'*S*' é verdadeira sse *p*	Conhecimento semântico +
'*S*' é verdadeira.	Conhecimento do valor de verdade =
logo, *p*	Conhecimento de mundo (fatos)

1.2 Relações semânticas entre sentenças

Passemos agora a um outro tipo de intuição semântica. Como falantes do português, todos nós temos intuições a respeito de certas relações entre sentenças, relações essas que, de um ponto de vista ainda pré-teórico, parecem envolver o significado dessas sentenças. Assim sendo, seria desejável que uma teoria semântica auxiliasse na definição e explicação dessas relações.

Considere, por exemplo, o par abaixo:

(2) a. João comprou um carro usado.
b. João comprou um carro.

Esse par ilustra a relação de ACARRETAMENTO. A ideia por trás dessa noção é a de que uma sentença S_1 acarreta uma sentença S_2 se a verdade de S_1 já trouxer embutida a verdade de S_2. Diz-se também que S_2 é uma consequência lógica de S_1.

Uma semântica baseada em condições de verdade nos permite definir precisamente essa noção: dizemos que uma sentença S_1 acarreta uma sentença S_2 se, sempre que S_1 for verdadeira, S_2 também será verdadeira. Ou, se quisermos empregar explicitamente a noção de CONDIÇÕES DE VERDADE, podemos dizer que uma sentença S_1 acarreta uma sentença S_2, se, sempre que as condições de verdade de S_1 forem satisfeitas (e S_1 for, portanto, verdadeira), as condições de verdade de S_2 também serão satisfeitas (e S_2 será, portanto, verdadeira).

Assim, dizemos que (2a) acarreta (2b), já que, se garantirmos a verdade da primeira, a verdade da segunda fica automaticamente garantida. Note que não necessitamos saber se as sentenças em questão são, de fato, verdadeiras ou não. Basta imaginarmos situações possíveis, reais ou não, em que uma das sentenças seja verdadeira e verificar se, em cada uma delas, a outra sentença é ou não verdadeira. Se existir ao menos uma situação em que S_1 é verdadeira, mas S_2 não, então S_1 não acarreta S_2.

Voltando às sentenças vistas acima, já notamos que (2a) acarreta (2b). As condições de verdade de (2a) nos dizem que a sentença será verdadeira se, e somente se, João tiver comprado um carro usado. Já as condições de verdade de (2b) nos dizem que a sentença será verdadeira se, e somente se, João tiver comprado um carro. Mas como quem compra um carro usado necessariamente compra um carro, então, em toda situação em que João tiver comprado um carro usado ele terá comprado um carro. Note, entretanto, que o inverso não é verdadeiro. Ou seja, alguém que comprou um carro não necessariamente comprou um carro usado. Basta imaginar uma situação em que o carro em questão é novo. Por-

tanto, (2b) não acarreta (2a). Veremos mais adiante neste capítulo que, se formos explícitos na formulação das condições de verdade das sentenças de uma língua, o método que estamos começando a esboçar também nos auxiliará na explicação dessa nossa intuição tão clara (e mesmo banal) de que se alguém comprou um carro usado, então esse alguém necessariamente comprou um carro, mas não o contrário.

Passemos, agora, à noção de equivalência ou sinonímia entre sentenças. Dizemos de duas sentenças que elas são EQUIVALENTES (ou sinônimas), se as suas condições de verdade são as mesmas. Ou seja, não há como satisfazer as condições de verdade de uma sem automaticamente satisfazer as da outra. Por exemplo, (3a) e (3b) abaixo são equivalentes. Não há como imaginar uma situação em que uma é verdadeira e a outra falsa, e vice-versa.

(3) a. João visitou Maria.
    b. Maria foi visitada por João.

Note que é possível definir equivalência em termos de acarretamento. S_1 e S_2 são equivalentes se, e somente se, S_1 acarreta S_2 e S_2 acarreta S_1.

Por fim, mais algumas noções relevantes que se aplicam a sentenças e que são definidas em termos de condições de verdade. Dizemos que uma sentença é uma TAUTOLOGIA se as suas condições de verdade forem sempre satisfeitas, não havendo situação possível em que ela seja falsa. É o caso, por exemplo, da sentença abaixo:

(4) Ou João comprou um carro ou João não comprou um carro.

Por mais que sejamos criativos, jamais conseguiremos imaginar uma situação em que essa sentença seja falsa. Podemos também pensar em sentenças que nunca são verdadeiras. Suas condições de verdade não são jamais satisfeitas. Nesse caso dizemos que a sentença em questão é uma CONTRADIÇÃO. Um exemplo é a sentença abaixo:

(5) O Brasil tem mais habitantes do que Portugal, e Portugal tem mais habitantes do que o Brasil.

Restam as sentenças cujas condições de verdade podem ou não ser satisfeitas, dependendo da situação (fatos) em questão. Essas sentenças são chamadas de CONTINGÊNCIAS. É o caso, por exemplo, da sentença "João comprou um carro usado".

1.3 Composicionalidade

Voltemos a considerar nossa intuição sobre o significado de uma sentença. Intuitivamente, o significado de uma sentença depende do significado dos itens lexicais que a compõem. Além de depender do significado dos itens lexicais que a compõem, o significado de uma sentença depende também da maneira como esses itens se encontram organizados na sentença. O significado de "João beijou Maria" difere não só do significado de "João abraçou Maria", mas também do significado de "Maria beijou João".

Para captar essas intuições, precisamos dizer algo sobre a relação entre o significado de uma sentença e o significado das partes dessa sentença. Isso parece necessário, já que o contrário nos levaria a assumir que nossa competência semântica se reduz a uma lista de sentenças associadas a suas condições de verdade. Mas como explicaríamos então nossa capacidade de compreender sentenças que nunca ouvimos antes? (6) abaixo talvez seja um caso em questão:

(6) Uma tartaruga com quatro estômagos engoliu a mãe de um elefante cor de abóbora.

Um falante competente, mesmo que se depare com a sentença acima pela primeira vez e fora de qualquer contexto relevante (por exemplo, escrita em um muro), será capaz de imaginar cenários hipotéticos que tornariam a sentença verdadeira, por mais bizarros que sejam.

E, se refletirmos melhor, veremos que somos capazes de compreender um número infinito de sentenças, ainda que na prática isso não seja óbvio devido a restrições de memória e processamento. Basta pensar em uma série de sentenças como a seguinte:

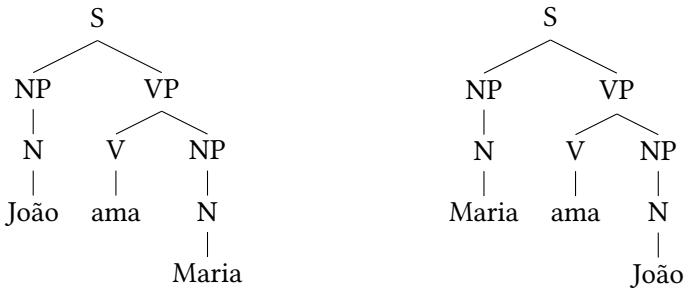
(7) a. O João era poeta.
    b. O pai do João era poeta.
    c. O pai do pai do João era poeta.
    d. O pai do pai do pai do João era poeta.
    e. …

Note que, mesmo com um léxico bastante reduzido (6 itens), somos capazes não só de reconhecer como gramaticais, mas também de interpretar um número infinito de sentenças. A hipótese de uma lista infinita armazenada em um cérebro finito não parece consistente. Nossa competência semântica é portanto mais do que uma simples lista de sentenças pareadas com suas condições de verdade. Em

consonância com essas intuições, vamos adotar o seguinte princípio:

> **Princípio de composicionalidade**:
> O significado de uma sentença é derivado do significado dos itens lexicais que a compõem e da maneira como esses itens estão organizados.

Para representar a maneira como os itens lexicais de uma sentença estão organizados, ou seja, para representar a estrutura sintática de uma sentença, vamos nos valer de diagramas em forma de árvore. Para sentenças simples, assumiremos que são formadas por um sintagma nominal (NP, do inglês *noun phrase*) e um sintagma verbal (VP, do inglês *verb phrase*). Sintagmas verbais, por sua vez, são formados por um verbo intransitivo, ou por um verbo transitivo seguido de um sintagma nominal, como mostrado a seguir:



Note que apesar de conterem os mesmos itens lexicais, as estruturas acima são distintas, já que seus constituintes (as sub-árvores que as compõem) não são os mesmos. Por exemplo, o constituinte correspondente ao nó rotulado de VP engloba (domina) o item lexical *Maria* na primeira estrutura, mas não na segunda. À medida que progredirmos, apresentaremos as estruturas de sentenças mais complexas.

Tendo estruturas como essas em mente, suponha agora que a noção de significado se aplique não apenas a itens lexicais, mas a todo e qualquer constituinte sintático (incluindo, claro, sentenças). Podemos, então, formular uma versão mais específica do princípio acima, que continuaremos a chamar simplesmente de princípio de composicionalidade.

O significado de um constituinte sintático é derivado do significado de seus constituintes imediatos.

O Princípio se aplica tanto a sentenças quanto a constituintes menores, como sintagmas nominais, projeções verbais, etc. De certa forma, trata-se de uma interpretação radical da noção de composicionalidade, já que para obter o significado de um constituinte, basta olhar para o significado de seus constituintes imediatos, sem se preocupar com o significado das partes que compõem esses constituintes. O resultado disso é que podemos vislumbrar um procedimento recursivo, que computa o significado de um constituinte *C*, baseado nos significados de seus subconstituintes. Por sua vez, o significado de cada um desses constituintes (D_1,...,D_n) é obtido a partir do significado de seus próprios subconstituintes E_1,...,E_n ("sub-subconstituintes" de *C*), e assim por diante, descendo até os itens lexicais, que teriam seus significados armazenados em uma lista (um léxico).

Como ilustração, considere a estrutura apresentada mais acima que corresponde à sentença *Maria ama João*. Nosso objetivo é obter o significado de S (ou, mais precisamente, da estrutura rotulada de S). De acordo com o princípio de composicionalidade, o significado de S é obtido a partir do significado de NP_1 e do significado de VP. O significado de NP_1 é obtido a partir do significado de N_1, que por sua vez é obtido do significado do item lexical *Maria*. Já o significado de VP é obtido a partir do significado de V e do significado de NP_2. O significado de V é obtido a partir do significado do verbo *ama* e o significado de NP_2 a partir do significado de N_2, que por sua vez é obtido a partir do significado de *João*. Tudo o que precisamos agora é de um lista com as entradas correspondentes ao significado dos itens lexicais. Feito isso, poderíamos percorrer o caminho inverso ao que acabamos de fazer e obter o significados de N_2, de NP_2, de V, de VP, de N_1, de NP_1 e, finalmente, de S!

Obviamente, dizer que o significado de um constituinte é derivado do significado de suas partes ainda deixa no ar a questão de como esse significado é obtido. Falta-nos ainda especificar as regras ou princípios de composição. Serão essas regras que nos dirão explicitamente de que maneira do significado das partes obtém-se o significado do todo.

Tudo isso, claro, ainda é muito programático, mas já sabemos que formato nossa teoria semântica terá se seguirmos a versão do princípio de composicionalidade discutida acima: um léxico, ou seja, uma lista com o significado das palavras da língua em questão e um conjunto de regras composicionais, definindo explicitamente como obter o significado de um constituinte a partir de seus subconstituintes imediatos.

É com a definição dessas regras e com a elaboração de entradas lexicais para uma série de palavras de diferentes classes gramaticais que nos ocuparemos a

partir do próximo capítulo, tendo sempre como objetivo a derivação de condições de verdade condizentes com nossa intuição a respeito do significado das sentenças analisadas.

1.4 Linguagem objeto e metalinguagem

Daqui em diante, nosso interesse estará voltado para as sentenças do português, ainda que os métodos utilizados se apliquem igualmente a outras línguas naturais (essa pelo menos é nossa esperança). O português será portanto nossa LINGUAGEM OBJETO, a linguagem sobre a qual iremos teorizar. Precisamos definir ainda uma METALINGUAGEM, a linguagem na qual iremos formular nossa teoria. Em particular, precisamos de uma linguagem adequada para representar as condições sobre o mundo que aparecem à direita do conectivo *se e somente se* nas representações dos significados das sentenças.

***S* é verdadeira se e somente se p**

Utilizaremos uma mistura de português com alguns símbolos da lógica de predicados e da teoria dos conjuntos, sempre almejando obter representações sucintas e sem ambiguidades. O fato de utilizarmos o próprio português em nossa metalinguagem poderá, às vezes, causar a impressão de que nossa teoria está apenas relatando o óbvio, como no caso abaixo:

Está chovendo é verdadeira se e somente se está chovendo.

Tenha em mente que isso é ilusório. Fosse nossa linguagem objeto o inglês, a ilusão desapareceria:

It is raining é verdadeira se e somente se está chovendo.

Note que, nesse segundo caso, não se trata de uma afirmação óbvia. Note, ainda, que tampouco se trata de uma tradução do inglês para o português. O que aparece do lado direito de *se e somente se* diz respeito ao mundo, e não a uma sentença do português. Em outras palavras, as afirmações acima não são equivalentes às afirmações abaixo:

Está chovendo é verdadeira se e somente se *Está chovendo* é verdadeira.

It is raining é verdadeira se e somente se *Está chovendo* é verdadeira.

Note que, nesses casos, estaríamos relacionando linguagem e linguagem, e não linguagem e mundo. Ou seja, fazer semântica não se limita a apresentar paráfrases ou traduções. Nunca será demais relembrar que sempre que estivermos diante de uma representação da forma

***S* é verdadeira se e somente se p**

o que aparece do lado direito do conectivo *se e somente se* é uma descrição de um estado de coisas, que, acidentalmente, pode estar codificado na própria linguagem objeto.

1.5 Extensões

Assumimos acima que o significado de uma sentença corresponde a suas condições de verdade. Nossa teoria especifica as condições que devem ser satisfeitas para que uma sentença seja verdadeira. Quando uma sentença é verdadeira, dizemos que seu valor de verdade é 1 (verdadeiro). Quando uma sentença é falsa, dizemos que seu valor de verdade é 0 (falso). Diremos de uma sentença verdadeira que sua EXTENSÃO ou DENOTAÇÃO é 1. Diremos de uma sentença falsa que sua extensão ou denotação é 0. Utilizaremos um símbolo especial para o termo *extensão*: ⟦ ⟧. Além disso, abreviaremos o conectivo *se e somente se* como *sse*. As afirmações abaixo são, portanto, equivalentes:

S é verdadeira se, e somente se, p

⟦S⟧ = 1 sse p

Note que a extensão de uma sentença não é o seu significado. Conforme já notamos no início, saber o significado de uma sentença não é o mesmo que saber se a sentença é verdadeira ou falsa.

Podemos interpretar o conceito de extensão de uma sentença como sendo uma função (no sentido matemático do termo) que mapeia a representação sintática de uma sentença em um valor de verdade. Uma teoria semântica se incumbe então de especificar as condições para que uma sentença S qualquer seja mapeada no valor de verdade 1.

Vamos aplicar o conceito de extensão não só a sentenças, mas a constituintes

sintáticos em geral. A ideia é atribuir a cada um deles (incluindo os itens lexicais) um objeto que funcione como seu valor semântico, ou seja, algo manipulável pelas regras de composição no percurso para a obtenção das condições de verdade de uma sentença. Isso nos leva a uma versão extensional do princípio de composicionalidade:

A extensão de um constituinte sintático é derivada da extensão de seus constituintes imediatos.

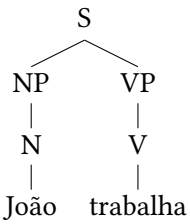
Uma teoria semântica é dita EXTENSIONAL se suas regras de composição especificam como obter a extensão de um constituinte a partir das extensões de seus subconstituintes.

Tudo isso ainda está sendo colocado de forma muito abstrata e ficará mais claro e concreto à medida que formos progredindo na análise de sentenças específicas. Mas, a título de ilustração provisória, vamos considerar um pequeno grupo de sentenças, criando um sistema composicional rudimentar pautado no conceito de extensão. Vamos nos valer de algumas ferramentas matemáticas bem simples retiradas da teoria dos conjuntos ensinada nos ensinos fundamental e médio.

Comecemos pela sentença abaixo:

(8) João trabalha.

Essa sentença é formada por um sintagma nominal (NP) funcionando como sujeito e um sintagma verbal (VP) funcionando com predicado. O NP sujeito é formado exclusivamente por um nome (N), no caso o nome próprio *João*. O VP é formado exclusivamente por um verbo (V), no caso o verbo intransitivo *trabalhar*. Tudo isso está representado na estrutura abaixo, que servirá de entrada para o sistema interpretativo que estamos criando:



Vamos proceder de baixo para cima, começando pelos itens lexicais. Para o nome próprio, vamos assumir simplesmente que sua extensão é o indivíduo (em carne e osso) portador do nome (faça de conta, para simplificar, que só exista um

João no mundo):

⟦João⟧ = o indivíduo João

Ou simplesmente:

⟦João⟧ = João

Com isso, estamos assumindo que saber o significado de um nome próprio é simplesmente saber a identidade do portador do nome. Ou seja, no caso dos nomes próprios, o sentido é a própria extensão. Já no caso do verbo *trabalhar*, não queremos associá-lo a indivíduos específicos, mas sim a uma classe ou conjunto de indivíduos, a saber, o conjunto dos indivíduos que trabalham.

⟦trabalha⟧ = $\{x|x$ trabalha$\}$

A notação $\{x|\phi\}$ deve ser lida como o conjunto formado única e exclusivamente pelos objetos x que satisfazem a condição ϕ.

Note que, nessa representação da extensão do verbo, evitamos menção a indivíduos particulares na especificação do conjunto. Ao contrário, optamos pela especificação de uma condição: para que um indivíduo *x* qualquer pertença à extensão desse verbo é necessário (e suficiente) que este indivíduo trabalhe. Com isso, queremos salientar que a extensão de um verbo não equivale a seu significado. De fato, saber o significado do verbo *trabalhar* não implica saber quem são nem quantas são as pessoas que trabalham no mundo. Ao contrário, saber o significado desse verbo implica apenas saber quais as condições que um indivíduo *x* qualquer deve satisfazer para que se possa dizer de *x* que *x* trabalha. Assim, a entrada lexical acima ganha plausibilidade cognitiva, já que mostra que a extensão do verbo é um conjunto de indivíduos, sem, no entanto, especificar a identidade desses indivíduos. Em suma, apesar de estarmos lidando com um sistema extensional, não estamos nos comprometendo com a ideia de que o significado seja equivalente à extensão. Já havíamos visto isso com as sentenças, e agora estamos vendo um ponto semelhante com os verbos.

Feitas essas observações, continuemos com a interpretação de (8). Comecemos pelo NP sujeito. Como se trata de um nó não ramificado, vamos assumir que ele herda a extensão do nó imediatamente dominado por ele, ou seja, N:

$$\left[\!\!\left[\begin{array}{c} \text{NP} \\ | \\ \text{N} \\ | \\ \text{João} \end{array} \right]\!\!\right] = \left[\!\!\left[\begin{array}{c} \text{N} \\ | \\ \text{João} \end{array} \right]\!\!\right]$$

Como **N** também é um nó não ramificado, temos:

$$\left[\!\!\left[\begin{array}{c} \text{N} \\ | \\ \text{João} \end{array} \right]\!\!\right] = [\![\text{João}]\!]$$

João é um item lexical e, para sabermos sua extensão, basta consultarmos a entrada lexical correspondente:

$[\![\text{João}]\!] = \text{João}$

Em resumo:

$$\left[\!\!\left[\begin{array}{c} \text{NP} \\ | \\ \text{N} \\ | \\ \text{João} \end{array} \right]\!\!\right] = \left[\!\!\left[\begin{array}{c} \text{N} \\ | \\ \text{João} \end{array} \right]\!\!\right] = [\![\text{João}]\!] = \text{João}$$

Efetuando um cálculo análogo para o sintagma verbal, obteremos:

$$\left[\!\!\left[\begin{array}{c} \text{VP} \\ | \\ \text{V} \\ | \\ \text{trabalha} \end{array} \right]\!\!\right] = \left[\!\!\left[\begin{array}{c} \text{V} \\ | \\ \text{trabalha} \end{array} \right]\!\!\right] = [\![\text{trabalha}]\!] = \{x | x \text{ trabalha}\}$$

Para simplificar a representação de nossas derivações semânticas, em vez de colocarmos árvores inteiras no interior do símbolo de extensão, utilizaremos apenas o rótulo correspondente à raiz da estrutura. Assim, os resultados que acabamos de obter podem ser apresentados da seguinte forma:

$[\![\text{NP}]\!] = [\![\text{N}]\!] = [\![\text{João}]\!] = \text{João}$

$[\![\text{VP}]\!] = [\![\text{V}]\!] = [\![\text{trabalha}]\!] = \{x | x \text{ trabalha}\}$

Resta-nos agora calcular a extensão de S, a partir das extensões de NP e VP. A intuição é que para que a sentença (S) seja verdadeira, a extensão do sujeito (NP) deve pertencer à extensão do predicado (VP):

⟦S⟧ = 1 *sse* ⟦NP⟧ ∈ ⟦VP⟧

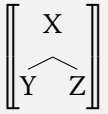
⟦S⟧ = 1 *sse* João ∈ $\{x|x$ trabalha$\}$

Em outras palavras:

⟦S⟧ = 1 *sse* João trabalha

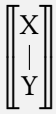
Esse é exatamente o resultado que queríamos obter. A condição necessária e suficiente para que a sentença (8) seja verdadeira é que o João trabalhe. Precisamos apenas explicitar os mecanismos ou regras que usamos na derivação do resultado. Vamos chamá-los de Regra I e Regra II:

Regra I:
Seja X um nó ramificado, cujos constituintes imediatos são Y e Z. Se ⟦Y⟧ é um indivíduo e ⟦Z⟧ é um conjunto, então:



= 1 *sse* ⟦Y⟧ ∈ ⟦Z⟧

Regra II:
Seja X um nó não ramificado, cujo único constituinte imediato é Y, então:

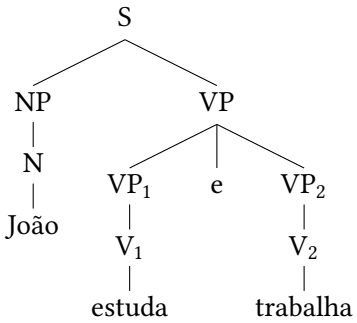


= ⟦Y⟧

Resumindo: Obtivemos as condições de verdade de uma sentença valendo-nos exclusivamente de entradas lexicais e regras composicionais explicitamente formuladas, dentro, portanto, do espírito composicional e extensional que apresentamos mais acima.

Vejamos agora mais um exemplo, que servirá mais tarde para discutirmos o formalismo que estamos adotando. Considere a sentença (9) abaixo, para a qual vamos assumir a estrutura a seguir:

(9) João estuda e trabalha.



Para os nós não ramificados da estrutura acima, já sabemos como proceder por meio da Regra II:

$[\![\text{NP}]\!] = [\![\text{N}]\!] = [\![\text{João}]\!] = \text{João}$

$[\![\text{VP}_1]\!] = [\![\text{V}_1]\!] = [\![\text{estuda}]\!] = \{x \mid x \text{ estuda}\}$

$[\![\text{VP}_2]\!] = [\![\text{V}_2]\!] = [\![\text{trabalha}]\!] = \{x \mid x \text{ trabalha}\}$

Para a conjunção *e*, vamos assumuir que sua extensão é a operação de interseção de conjuntos:

$[\![\text{e}]\!] = \cap$

Dados dois conjuntos A e B, a interseção de A e B ($A \cap B$) é o conjunto formado pelos elementos que pertencem tanto a A quanto a B. Passemos, então, à interpretação do sintagma verbal VP, que tem três constituintes imediatos. Dois deles têm por extensão conjuntos, e o outro tem por extensão a operação de interseção. A intuição é que, em casos assim, aplicamos a operação aos conjuntos. Vamos explicitar esse procedimento através de uma outra regra:

Regra III:
Seja X um nó ramificado. Se a extensão de um dos constituintes imediatos de X for uma operação sobre conjuntos e as extensões dos demais constituintes imediatos forem conjuntos, então ⟦X⟧ é o conjunto resultante da aplicação daquela operação a esse(s) conjunto(s).

Voltando ao exemplo (9), podemos aplicar a Regra III para obter a extensão de VP:

⟦VP⟧ = ⟦VP_1⟧ ∩ ⟦VP_2⟧

⟦VP⟧ = $\{x|x$ estuda$\} \cap \{x|x$ trabalha$\}$

⟦VP⟧ = $\{x|x$ estuda e x trabalha$\}$

Como a Regra II não faz menção à ordem linear dos constituintes em questão, poderíamos igualmente ter escrito:

⟦VP⟧ = ⟦VP_2⟧ ∩ ⟦VP_1⟧

No caso acima, isso não tem maiores consequências, já que a operação de interseção é comutativa, ou seja, $A \cap B$ é equivalente a $B \cap A$.

De volta à nossa derivação, devemos agora calcular a extensão de S em função das extensões de NP e VP. Podemos aqui aplicar a Regra I:

⟦S⟧ = 1 *sse* ⟦NP⟧ ∈ ⟦VP⟧

⟦S⟧ = 1 *sse* João ∈ $\{x|x$ estuda e x trabalha$\}$

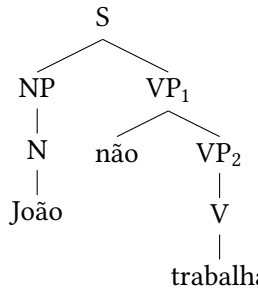
Em palavras:

⟦S⟧ = 1 *sse* João estuda e João trabalha.

Novamente, obtivemos o resultado desejado: a condição necessária e suficiente para que a sentença (9) seja verdadeira é que João estude e trabalhe.

Por fim, passemos à interpretação da sentenaça (10) abaixo, cuja estrutura apresentamos em seguida:

(10) João não trabalha.



A novidade aqui é a negação *não*, que representamos como um adjunto adverbial. A exemplo da conjunção *e*, vamos interpretá-la como uma operação sobre conjuntos. Nesse caso, trata-se da operação de complementação. Dado um conjunto A, o complemento de A ($\overline{A}$) é o conjunto dos elementos que não pertencem a A. Temos então:

$[\![\text{não}]\!]$ = $\overline{}$ (operação de complementação)

Já para VP_2, teríamos:

$[\![\text{VP}_2]\!] = [\![\text{V}]\!] = [\![\text{trabalha}]\!] = \{x | x \text{ trabalha}\}$

Para VP_1, já temos tudo o que precisamos: trata-se de um nó ramificado cujos constituintes têm por extensão um conjunto e uma operação. Basta, então, aplicarmos a Regra III:

$[\![\text{VP}_1]\!] = \overline{[\![\text{VP}_2]\!]}$

Pela definição da operação de complementação, teremos:

$[\![\text{VP}_1]\!] = \{x | x \notin [\![\text{VP}_2]\!]\}$

$[\![\text{VP}_1]\!] = \{x | x \notin \{x | x \text{ trabalha}\}\}$

$[\![\text{VP}_1]\!] = \{x | x \text{ não trabalha}\}$

Chegamos, por fim, à relação sujeito-predicado no topo da sentença. Basta aplicarmos a regra I:

$[\![S]\!]$ = 1 *sse* $[\![NP]\!] \in [\![VP_1]\!]$

$[\![S]\!]$ = 1 *sse* joão $\in \{x | x$ não trabalha$\}$

$[\![S]\!]$ = 1 *sse* João não trabalha.

Uma vez mais, obtivemos o resultado desejado: a condição necessária e suficiente para que a sentença (10) seja verdadeira é que João não trabalhe.

É comum apresentar uma derivação semântica com o uso de linhas numeradas e a indicação a cada passo da justificativa do que está sendo afirmado. Para essa estrutura que acabamos de analisar, teríamos:

1. $[\![NP]\!] = [\![N]\!] = [\![João]\!]$ = João (Regra II: 2x; Léxico)

2. $[\![VP_2]\!] = [\![V]\!] = [\![estuda]\!] = \{x \mid x$ trabalha$\}$ (Regra II: 2x; Léxico)

3. $[\![não]\!] = \neg$ (Léxico)

4. $[\![VP_1]\!] = \neg[\![VP_2]\!]$ (Linha 3; Regra III)

5. $[\![VP_1]\!] = \{x \mid x \notin [\![VP_2]\!]\}$ (Linha 4 e definição de complemento)

6. $[\![VP_1]\!] = \{x \mid x \notin \{x \mid x$ trabalha$\}\}$ (Linhas 2 e 5)

7. $[\![VP_1]\!] = \{x$: *x* não estuda$\}$ (Linha 6)

8. $[\![S]\!]$ = 1 *sse* $[\![NP]\!] \in [\![VP_1]\!]$ (Regra I)

9. $[\![S]\!]$ = 1 *sse* João $\in \{x \mid x$ *não trabalha*$\}$ (Linhas 1, 7 e 8)

10. $[\![S]\!]$ = 1 *sse* João não trabalha. (linha 9 e definição de conjunto)

O alcance empírico do sistema que acabamos de ilustrar é obviamente bastante limitado, já que nosso léxico é constituído por apenas cinco itens. Ainda assim, o poder do método que vimos acima é capaz de fornecer explicações interessantes a respeito de algumas intuições semânticas que todo falante do português tem sobre as sentenças (8)-(10), repetidas abaixo em (11)-(13):

(11) João trabalha.

(12) João estuda e trabalha.

(13) João não trabalha.

Por exemplo, todos nós concordamos que se (12) for verdadeira, então (11) será necessariamente verdadeira e (13) necessariamente falsa. Vejamos o que nosso sistema tem a nos dizer: de acordo com o que vimos acima, assumir que (12) é verdadeira é o mesmo que afirmar que o indivíduo João (que representaremos abaixo por *j*) pertence à interseção do conjunto das pessoas que trabalham (vamos representar esse conjunto por *T*) com o conjunto das pessoas que estudam (vamos representar esse conjunto por *E*). Na notação da teoria dos conjuntos:

$$\mathrm{j} \in (\mathrm{T} \cap \mathrm{E}) \tag{i}$$

Mas sabemos pelas definições da teoria dos conjuntos que se um elemento *x* qualquer pertence à interseção de dois conjuntos *A* e *B* quaisquer, então *x* pertence a *A* e *x* pertence a *B*. Logo, tanto (ii) quanto (iii) abaixo são consequências de (i):

$$\mathrm{j} \in \mathrm{T} \tag{ii}$$

$$\mathrm{j} \in \mathrm{E} \tag{iii}$$

Mas, de acordo com o que vimos mais acima, assumir (ii) é o mesmo que assumir que a sentença (11) é verdadeira. Temos assim uma explicação simples e elegante sobre nossa intuição a respeito do fato de (12) acarretar (11). Na mesma linha, a teoria dos conjuntos nos informa que para qualquer conjunto A, um elemento x qualquer ou pertence a A ou não pertence a A, mas nunca as duas coisas ao mesmo tempo. Sendo assim, (ii) contradiz (iv) abaixo:

$$\mathrm{j} \notin \mathrm{E} \tag{iv}$$

Mas vimos acima que assumir (iv) é o mesmo que afirmar que (13) é verdadeira. Logo, a verdade de (12) implica na falsidade de (13). Novamente, temos uma explicação precisa sobre nossa intuição a respeito da relação semântica entre essas duas sentenças.

Cumprimos assim o principal objetivo deste capítulo: motivar a construção de um sistema interpretativo vericondicional que atribui condições de verdade

para as sentenças de uma língua baseado no valor semântico (extensões) de suas partes (os constituintes sintáticos). Motivamos ainda o uso de objetos matemáticos na representação destes valores semânticos, indicando como uma abordagem formal pode lançar luz sobre nossas capacidade de realizar inferências de cunho semântico.

Continuaremos a proceder dessa forma no restante deste livro, incrementando gradualmente o alcance empírico da teoria através da análise de uma série de construções linguísticas. Quanto ao formalismo, faremos uma pequena mudança de perspectiva que nos permitirá um poder de generalização maior na formulação das regras de composição. Como o leitor notará já a partir do próximo capítulo, ao invés de lidarmos diretamente com conjuntos, manipularemos FUNÇÕES. De fato, o uso de funções será tão frequente no restante do livro, que iniciaremos o próximo capítulo com uma discussão sobre elas.

1.6 Além de extensões e condições de verdade

A identificação da extensão de uma sentença com seu valor de verdade e de seu significado com suas condições de verdade pode ser taxada de limitada por uma série de motivos. Nesta seção vamos olhar para alguns deles. Reconhecer essas limitações e refletir sobre suas consequências servirá para tornar mais claro nosso real objeto de estudo.

Em primeiro lugar, considere os exemplos abaixo:

(14) a. A Terra é plana.
b. Pedro acredita que a Terra é plana.

Note que, apesar de (14b) ter (14a) como um de seus constituintes sintáticos, o valor de verdade de (14b) não depende do valor de verdade de (14a). Pode-se acreditar em coisas falsas, bem como não acreditar em coisas verdadeiras. Lembre-se, entretanto, de que a extensão de uma sentença é seu valor de verdade e que, de acordo com a versão extensional do princípio de composicionalidade que assumimos, a extensão de um constituinte é calculada a partir da extensão de seus subconstituintes.

Exemplos como os acima colocam em xeque o sistema extensional que adotamos. Precisamos acessar o significado da oração subordinada, e não seu valor de verdade, para calcular o valor de verdade da oração principal. Para tanto, torna-se necessário atribuir a sentenças e a constituintes sintáticos em geral um outro tipo de valor semântico que possa ser manipulado pelas regras composicionais

e que capte o seu significado. Esse outro valor semântico costuma ser chamado de INTENSÃO (com a letra *s*). No caso de sentenças declarativas, a intensão é chamada de PROPOSIÇÃO. Assume-se, assim, que o verbo *acreditar* em (14b) relaciona indivíduos e proposições.

É importante salientar que não se trata, simplesmente, de abandonar o sistema extensional que delineamos neste capítulo e que desenvolveremos no restante deste livro. Precisamos, sim, suplementá-lo com ingredientes intensionais a fim de analisarmos sentenças como (14b), por exemplo. Mas, como veremos a partir do próximo capítulo, há uma série de construções linguísticas para as quais um sistema puramente extensional é bem sucedido.

Passando ao próximo ponto, ser ou não ser verdadeira não é um atributo que pareça apropriado a sentenças não declarativas, como as abaixo:

(15) a. Quem o João está namorando?
b. O João está namorando?
c. Saia da sala!
d. Passe a salada, por favor.

(15a) e (15b) são sentenças interrogativas que exprimem pedidos de informação. Não faz sentido perguntar se são verdadeiras ou falsas. Por conseguinte, não faz sentido atribuir a elas condições de verdade. O mesmo se pode dizer das sentenças imperativas em (15c) e (15d), que exprimem uma ordem e um pedido, respectivamente. Se essa crítica atinge em cheio nossa máxima de que saber o significado de uma sentença é saber suas condições de verdade, devemos nos perguntar se ela deve afetar também nosso propósito de adotar um sistema composicional extensional dentro das linhas sugeridas na seção anterior. A resposta aqui é negativa. A única conclusão a que devemos chegar frente a essa crítica é que a extensão de uma sentença não declarativa não é um valor de verdade e seu significado não se reduz a condições de verdade. Aqui podemos apenas especular que isso se deve, ao menos em parte, à extensão e ao significado de palavras interrogativas, de formas verbais imperativas, ou mesmo de morfemas abstratos, não pronunciados, mas presentes na estrutura destas sentenças (o que tem um certo apelo frente a casos como (15b), por exemplo).

A lição que devemos tirar dessas considerações é que nossa máxima deve ser entendida de forma mais restrita: saber o significado de certas sentenças declarativas é saber suas condições de verdade. Mas nunca é demais reafirmar que em princípio isso em nada abala nossa abordagem composicional baseada na atribuição de extensões a constituintes sintáticos.

O mesmo se pode dizer a respeito de certas sentenças declarativas chamadas de performativas:

(16) a. Eu vos declaro marido e mulher.
b. Eu te batizo em nome do Pai, do Filho e do Espírito Santo.

Quando um padre pronuncia sentenças como essas, ele não está descrevendo um pedaço do mundo. Agindo de acordo com um ritual e com o poder a ele atribuído, suas próprias palavras têm valor de transformação, ou seja, é a própria enunciação da sentença que torna os noivos um casal perante a igreja, assim como a pessoa batizada um cristão. Nesses casos é o próprio ato de fala que está em questão, e de fato soa estranho perguntar a alguém presente em uma dessas cerimônias se o que o padre está dizendo é verdade ou não. Cabem aqui as mesmas observações que fizemos acima a respeito das sentenças em (15), não havendo razões para abandonarmos nossa abordagem composicional extensional. Devemos apenas limitar nossa meta inicial (a atribuição de condições de verdade) a um subconjunto das sentenças declarativas e refletir sobre que objetos atribuir a sentenças performativas. Novamente, uma discussão mais profunda sobre essas sentenças e o seu uso foge ao escopo deste livro (mas ver sugestões de leitura ao final do capítulo).

Por fim, devemos ter em mente que *significado* para nós é um termo técnico, que certamente não faz jus às diversas acepções que essa palavra tem no uso ordinário da língua. Isso se torna especialmente claro ao se notar que *significado* como viemos assumindo é algo que se aplica a sentenças isoladas e que não leva em conta aspectos relacionados ao seu uso em contextos específicos. Considere por exemplo os dois diálogos abaixo:

(17) A: Que tipo de roupa a namorada nova do João costuma usar?
B: Ela só usa roupas de grife.

(18) A: A namorada nova do João é bonita?
B: Ela só usa roupas de grife.

Nos dois casos, *B* responde a *A* com a mesma sentença. Mas fica claro que os significados (tomados no sentido ordinário do termo) dessas respostas não são os mesmos. No caso de (18), mas não no de (17), tendemos a inferir que *B* não acha a namorada nova do João bonita. A razão para essa diferença parece clara: em (18), *B* se esquiva de dar uma resposta direta à pergunta de *A*. Como não haveria razão para essa esquiva caso ele achasse a namorada de João bonita e

como pessoas tendem a ser polidas ao fazer julgamentos estéticos (sobretudo negativos), acabamos por concluir que *B* não acha a namorada de João bonita.

Note agora duas coisas. Em primeiro lugar, ao usarmos o termo *significado* no parágrafo anterior, nós o aplicamos à resposta de *B*. Uma resposta não é o mesmo que uma sentença. Sua caracterização faz menção necessária a um contexto de fala, sobretudo a uma pergunta que a antecede. Nesse sentido, fica claro porque as respostas de *B* acima têm significados diferentes: elas respondem perguntas distintas. Em segundo lugar, na origem das inferências acima está o fato de reconhecermos que, em (18), *B* se esquivou de dar uma resposta direta a *A*. Mas isso implica assumir que a sentença usada por *B* tem uma propriedade que independe do contexto em que é usada. Afinal, o que queremos dizer ao afirmar que em (17), mas não em (18), *B* dá uma resposta direta a *A*? Essa propriedade é justamente o *significado* da sentença usada por *B* em ambos os diálogos. Em suma, a compreensão do que se passa nos diálogos acima requer conhecimento do significado da sentença usada, bem como dos contextos em que ela é usada, incluindo aí as intenções do falante. Dessa interação resulta o que podemos chamar de significado do uso de uma sentença, SIGNIFICADO DO ENUNCIADO ou mesmo SIGNIFICADO DO FALANTE. Neste livro, é com o significado de sentenças que estaremos lidando, ainda que continuemos a usar o termo sem qualificativos. Essa distinção está no cerne da divisão de trabalho entre duas áreas da linguística preocupadas com a noção de significado entendida de maneira bastante ampla: a SEMÂNTICA, que estuda o significado literal ou convencional de sentenças (e suas partes) sem levar em conta o contexto de fala e as intenções do falante, e a PRAGMÁTICA, que estuda o significado de enunciados e que, portanto, não pode prescindir de uma caracterização precisa dos contextos em que usamos sentenças e outras expressões linguísticas, bem como dos diversos propósitos comunicativos subjacentes a esses usos.

Sugestões de leitura

Sobre os fundamentos de uma teoria semântica baseada em condições de verdade, as obras clássicas são Tarski (1944) e Davidson (1967). Introduções acessíveis podem ser encontradas em Taylor (1998), capítulo 3 e Larson & Segal (1995), capítulos 1 e 2. Quanto a seu enquadramento nas ciências cognitivas, ver Larson & Segal (1995), capítulo 13.

Sobre a ideia de se tomar um valor de verdade como sendo a extensão ou o referente de uma sentença, ver Frege (1892), outro grande clássico da área.

A grande figura que impulsionou os trabalhos atuais em semântica formal é o filósofo americano Richard Montague. Suas principais obras podem ser encontradas em Thomason (1974). Essas, entretanto, são obras de difícil leitura para o estudante que inicia seus estudos na área. Os leitores interessados poderão se beneficiar da ótima introdução em Gamut (1991), volume 2.

Sobre noções básicas de teoria dos Conjuntos, ver Partee et al. (1990), parte A. Sobre as relações entre diferentes acepções do termo significado, ver vários textos em Grice (1989). Dois grandes clássicos na área da pragmática em que aspectos não vericondicionais do significado são discutidos são Grice (1975) e Austin (1962). Para uma boa introdução à Pragmática, ver Levinson (1983), traduzido para o português como Levinson (2007), e Pires de Oliveira & Basso (2014). Para introduções à sintaxe, ver Carnie (2013), Haegeman (1994) e Mioto et al. (2005).

Por fim, gostaria de mencionar algumas introduções à semântica formal cujos conteúdos se superpõem parcialmente ao conteúdo deste livro: em inglês, Cann (1993), Larson & Segal (1995), Chierchia & McConnell-Ginet (2000), Heim & Kratzer (1998), Jacobson (2014), de Swart (2003), Winter (2016); em português, Chierchia (2003) e Pires de Oliveira (2001).

Exercícios

I. Dizer se (a) acarreta (b), e se (b) acarreta (a). Justifique sua resposta. Em particular, se sua resposta for NÃO, descreva explicitamente situações possíveis que suportem esta resposta.

(a) O menor estado brasileiro fica na região Nordeste.
(b) Sergipe fica na região Nordeste.

II. Justifique as seguintes afirmativas, em que *S* é uma sentença qualquer:

(a) Se *S* é uma tautologia, então qualquer sentença acarreta *S*.
(b) Se *S* é uma contradição, então *S* acarreta qualquer sentença.
(c) Se *S* é uma contradição, então nenhuma contingência acarreta *S*.

III. Dizer se a seguinte afirmação é verdadeira ou falsa, justificando:

"nenhuma sentença acarreta sua negação."

IV. Assuma as seguintes extensões para os sintagmas verbais *veio pra USP* e *veio pra USP de carro*:

⟦viajou⟧ = $\{x|x$ viajou$\}$
⟦viajou de carro⟧ = $\{x|x$ viajou de carro$\}$

Valendo-se da teoria dos conjuntos, explique porque (a) acarreta (b), mas (b) não acarreta (a):

(a) João veio pra USP de carro.
(b) João veio pra USP.

Dica: dados dois conjuntos A e B quaisquer, A é um subconjunto de B ($A \subseteq B$) se, e somente se, todo elemento que pertence a A também pertence a B.

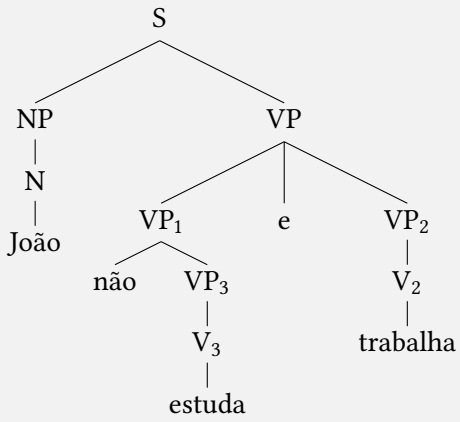
V. Calcule, passo a passo, as condições de verdade da sentença abaixo, de acordo com o sistema que desenvolvemos na seção 1.5:

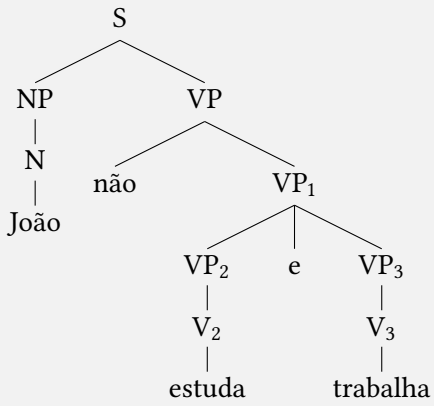
(a) João estuda e não trabalha.

VI. Considere a sentença abaixo:

(a) João não estuda e trabalha.

Podemos pensar em duas estruturas para essa sentença:





Para cada uma delas, calcule passo a passo as condições de verdade que o sistema da seção 1.5 fornece. Em seguida diga se alguma delas (ou ambas) condizem com suas intuições a respeito do significado dessa sentença.

VII. Considere a sentença abaixo:

(a) João estuda ou trabalha.

Assuma que o conectivo *ou* denote a operação de união (∪) entre conjuntos e calcule passo a passo as condições de verdade de (a). Discuta a adequação empírica do resultado, tomando por base sua intuição sobre o significado desta sentença. (Nota: um objeto x qualquer pertencerá à união entre dois conjuntos se, e somente se, x pertencer a pelo menos um desses conjuntos.)

2 Predicação

Neste capítulo, discutiremos como um predicado se combina semanticamente com seu(s) argumento(s). Olharemos primeiramente para sentenças com verbos intransitivos, transitivos e bitransitivos. Em seguida, passaremos a sentenças em que substantivos, adjetivos e preposições assumem a função de predicados principais. Quanto aos argumentos, limitaremos nossa atenção neste início aos nomes próprios.

2.1 Funções e a notação lambda

O objetivo desta seção inicial é familiarizar o leitor com o conceito matemático de FUNÇÃO e com a chamada NOTAÇÃO LAMBDA, que utilizaremos intensamente não apenas neste capítulo, mas em todo o livro. O leitor já familiarizado com esse material poderá apenas passar os olhos nesta seção para conferir algumas questões notacionais e terminológicas e já pular para a seção seguinte. Os demais devem ler atentamente o que se segue, já que isto será crucial na compreensão deste e dos demais capítulos.

Funções

Uma função pode ser vista como um dispositivo que efetua uma transformação sobre uma determinada classe de objetos. De acordo com essa caracterização mecânica ou dinâmica, uma função, ao receber determinado objeto (chamado de ARGUMENTO ou INPUT), retorna um outro objeto (chamado de VALOR ou OUTPUT). Para especificar uma função, precisamos informar o tipo de objetos que ela aceita como argumento, o tipo de objetos que ela retorna como valores e a descrição da transformação efetuada. Chamamos de DOMÍNIO da função o conjunto dos objetos que servem como argumento para a função. E de CONTRADOMÍNIO da função o conjunto que especifica o tipo de objeto que ela retorna como valor. Vejamos um exemplo. Seja f uma função que toma um número natural n qualquer e retorna o seu sucessor $n + 1$. Nesse caso, tanto o domínio quanto o contradomínio de f são o conjunto dos números naturais $\mathbb{N}$. Mas note que, como o número natu-

ral zero não é o sucessor de nenhum número natural, não há nenhum argumento que o retornará como valor. Chamamos de IMAGEM de uma função o subconjunto do contra-domínio que contém apenas os objetos que são efetivamente retornados como valor de algum argumento. No caso de *f*, a imagem é o conjunto dos números naturais positivos. É comum representar f da seguinte maneira:

$$\begin{aligned} f : \mathbb{N} &\to \mathbb{N} \\ x &\mapsto x + 1 \end{aligned}$$

Já na literatura sobre semântica, utiliza-se para a representação de funções uma notação criada pelo lógico americano Alonzo Church e conhecida como NOTAÇÃO LAMBDA em razão da utilização da letra grega de mesmo nome λ. Na notação lambda, nossa função f do parágrafo anterior seria representada da seguinte forma:

$$(\lambda n : n \in \mathbb{N}.\ n + 1)$$

De maneira mais geral, estaremos lidando com expressões da seguinte forma:

$$(\lambda v : \phi.\ \gamma)$$

Essa expressão representa a função que tem por domínio o conjunto dos elementos v que satisfazem ϕ e que transforma cada um desses elementos em γ. Nessa expressão, λ é chamado de OPERADOR LAMBDA, *v* é a variável ligada pelo operador lambda e γ é chamado de CORPO da expressão ou da função.

É prática comum simplificar a notação acima, omitindo-se a informação sobre o domínio da função, sempre que isso puder ser inferido do contexto, bem como os parênteses, quando isso não resultar em ambiguidade. Por exemplo, se adotarmos a convenção de usar a letra *n* para nos referirmos a números naturais, então a representação acima pode ser simplificada:

$$\lambda n.\ n + 1$$

Uma expressão do tipo $(\lambda v.\phi)$ é chamada de ABSTRAÇÃO-λ ou ABSTRAÇÃO FUNCIONAL. Diz-se que tal expressão foi formada a partir da expressão ϕ, abstraindo-se sobre a variável *v*. Nos casos que nos interessam aqui, a expressão ϕ contém pelo menos uma ocorrência da variável *v*. No nosso exemplo anterior, criamos a re-

presentação da função sucessor a partir da expressão $n + 1$, abstraindo sobre a variável *n*.

Para representarmos uma função ($\lambda v.\phi$) aplicada a um argumento *a* qualquer de seu domínio, utilizamos uma expressão como a abaixo:

$$(\lambda v.\ \phi)(a)$$

Uma expressão desse tipo é chamada de APLICAÇÃO FUNCIONAL ou simplesmente APLICAÇÃO. Retornando ao nosso exemplo, a aplicação da função sucessor ao número natural 4 é representada da seguinte forma:

$$(\lambda n.\ n + 1)(4)$$

A interpretação que desejamos para essa expressão é o sucessor do número 4, ou seja, o número 5. Para obtermos esse resultado, basta eliminarmos da representação da função o prefixo $\lambda n.$ e substituirmos a variável *n* pelo número 4 no corpo da função:

$$(\lambda n.\ n + 1)(4) \Rightarrow 4 + 1 \Rightarrow 5$$

Esse processo de simplificação, que corresponde intuitivamente à computação do valor de uma função aplicada a um argumento, é chamado de REDUÇÃO-β. Mas é comum também encontrarmos o termo CONVERSÃO-λ, que é o que adotaremos aqui.

Note ainda que a escolha da variável *n* nos exemplos acima foi, em um certo sentido, arbitrária. Poderíamos igualmente ter utilizado *m* ou *k*, por exemplo, que estaríamos ainda diante da mesma função:

$$(\lambda n.\ n + 1) \Leftrightarrow (\lambda m.\ m + 1) \Leftrightarrow (\lambda k.\ k + 1)$$

Esse processo de substituição da variável que segue o operador lambda e de todas as suas ocorrências no corpo da função por uma outra variável é chamado de REDUÇÃO-α ou CONVERSÃO-α. Tanto a conversão-α quanto a conversão-λ precisam estar sujeitas a determinadas condições para que produzam sempre os resultados desejados. Para saber mais sobre essas condições, ver o apêndice ao final do capítulo.

Conjuntos e funções características

Há um tipo especial de função que aparecerá com bastante frequência neste livro e para cuja representação a notação lambda também se mostra bastante adequada. Trata-se de funções que mapeiam cada um dos elementos de seu domínio ou no valor 0 ou no valor 1. Em outras palavras, o contradomínio dessas funções é o conjunto $\{0, 1\}$. Uma função desse tipo é chamada de FUNÇÃO CARACTERÍSTICA.

Para entender a razão desse nome, vamos trabalhar em cima de um exemplo concreto. Seja f uma função de $\mathbb{N}$ em $\{0, 1\}$ tal que $f(n) = 0$ se n for ímpar e $f(n) = 1$ se n for par. Na notação lambda, essa função pode ser representada da seguinte forma:

$\lambda n.\ n$ é par

A convenção é a seguinte: sempre que o corpo da função for uma sentença (afirmação), estamos diante de uma função característica. Tal sentença funciona como condição necessária e suficiente para um argumento ser mapeado no valor 1. Assim, temos que

$(\lambda n.\ n$ é par$)(3) = 0$ já que a sentença *3 é par* é falsa.

$(\lambda n.\ n$ é par$)(8) = 1$ já que a sentença *8 é par* é verdadeira.

De maneira geral, temos que:

$(\lambda n.\ n$ é par$)(m) = 1$ *sse* m for par.

Diz-se nesse caso que tal função caracteriza o conjunto dos números pares.

É importante ter em mente que lidar com conjuntos de elementos de um certo domínio ou com funções que levam elementos desse domínio em valores de verdade (funções características) é, essencialmente, a mesma coisa. Isso porque existe uma correspondência biunívoca (um a um) entre esses conjuntos e essas funções. Para cada função F com domínio D, pode-se definir o conjunto C (um subconjunto de D), de modo que, para todo x em D, tenhamos o seguinte:

$F(x) = 1\ sse\ x \in C$

Note, agora, que se dois conjuntos quaisquer C_1 e C_2 são distintos, então as funções correspondentes também serão distintas, já que haverá pelo menos um ele-

mento que pertencerá a apenas um dos conjuntos, e que, portanto, será mapeado no valor 1 por apenas uma das funções. Da mesma forma, a duas funções características distintas corresponderão sempre dois conjuntos distintos, já que pelo menos um elemento será mapeado por apenas uma delas no valor 1, e, portanto, pertencerá a apenas um dos conjuntos em questão. Em outras palavras, todo conjunto estará associado a uma única função característica, e toda função característica estará associada a apenas um único conjunto.

Dada essa correspondência biunívoca, podemos falar indistintamente em conjuntos ou em suas funções características sem perda de informação, escolhendo o que nos for mais conveniente. Esteja, portanto, preparado para conviver simultaneamente (às vezes em um mesmo texto) com jargões baseados em conjuntos e jargões baseados em funções.

Funções, conjuntos e predicados (propriedades)

Além dessa relação estreita entre funções características e subconjuntos de um domínio, tiraremos partido de uma outra conexão. Tome, por exemplo, o conjunto dos números pares e a função F que o caracteriza ($\lambda x.\ x$ é par). Dizemos que os elementos desse conjunto têm a propriedade de ser par, ou que o predicado PAR se aplica a esses elementos. A aplicação de um predicado P a um elemento x qualquer é representada pela expressão $P(x)$.

Temos então uma conexão tripla entre funções, conjuntos e predicados. Para deixar clara essa conexão nas representações que usaremos neste livro, partindo de uma função característica F qualquer, vamos representar seu conjunto característico por F_C e o predicado correspondente por F'. Dessa forma, dado um elemento x e uma função F quaisquer, teremos sempre a equivalência abaixo:

$$F(x) = 1 \leftrightarrow x \in F_C \leftrightarrow F'(x)$$

Tiraremos partido dessa dupla equivalência no decorrer de todo o livro, escolhendo a forma que nos for mais conveniente no momento. Em particular, a notação de predicados, por ser mais compacta, será usada com bastante frequência. Por fim, essa dupla equivalência também pode ser vista na sua forma negativa:

$$F(x) = 0 \leftrightarrow x \notin F_C \leftrightarrow \neg F'(x)$$

O símbolo $\neg$ (negação) à frente do predicado é oriundo da lógica. $\neg F'(x)$ significa que x não tem a propriedade F' ou que o predicado F' não se aplica a x.

Após esse breve tutorial, já estamos aptos a prosseguir na interpretação de estruturas que envolvem predicados e seus argumentos. Sempre que o leitor estiver inseguro quanto à parte formal da discussão, poderá retornar a esta seção até que esses conceitos, terminologia e notações estejam bem fixados.

2.2 Verbos intransitivos e aplicação funcional

No capítulo anterior, consideramos sentenças que têm um verbo intransitivo como predicado e um nome próprio como sujeito, como (1) abaixo:

(1) João trabalha.

Analisamos a extensão do nome próprio como sendo o indivíduo portador do nome e a extensão do verbo como o conjunto dos indivíduos que trabalham. A relação sujeito-predicado foi então analisada como pertinência: a sentença é verdadeira se, e somente se, a extensão do sujeito pertencer à extensão do predicado.

$[\![\text{João trabalha}]\!] = 1$ *sse* João $\in \{x | x \text{ trabalha}\}$.

$[\![\text{João trabalha}]\!] = 1$ *sse* João trabalha.

Com relação aos nomes próprios, manteremos o que já temos:

$[\![\text{João}]\!] = \text{João}$

Já com relação ao verbo *trabalhar*, efetuaremos uma mudança, que poderá parecer arbitrária de início, mas que nos permitirá um maior poder de generalização na formulação de nossas regras composicionais, conforme veremos ainda neste capitulo. Em vez de assumir que a extensão de um verbo intransitivo denota um conjunto, vamos assumir que a mesma é uma função, no sentido matemático do termo, como acabamos de ver na seção anterior. Mais precisamente, assumiremos que a extensão desses verbos é uma função característica, que leva indivíduos em valores de verdade, 0 ou 1. No caso do verbo *trabalhar*, por exemplo, sua extensão leva um indivíduo *x* no valor de verdade 1 se *x* trabalha e no valor de verdade 0 se *x* não trabalha. Usando a notação lambda, temos:

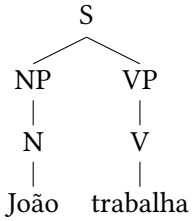
$[\![\text{trabalha}]\!] = \lambda x : x$ é um indivíduo. x trabalha

Assumindo que variáveis como *x*, *y*, *z* tenham sempre indivíduos como valores, podemos simplificar:

$[\![\text{trabalha}]\!] = \lambda x.\ x$ trabalha

Convém notar, antes de mais nada, que observações semelhantes às que fizemos a respeito da relação entre extensão e significado no capítulo anterior se aplicam a esse caso também. O significado do verbo *trabalhar* não é igual à sua extensão. Afinal, saber o significado desse verbo não é o mesmo que saber, para cada indivíduo *x*, se *x* trabalha ou não. Ao contrário, saber o significado deste verbo é saber as condições necessárias e suficientes para que se possa dizer de uma pessoa se é verdadeiro ou não que ela trabalha. Note, ainda, que a representação acima da extensão do verbo *trabalhar* faz menção a essas condições.

Olhemos agora para a representação sintática de nossa sentença.



Já sabemos a extensão dos itens lexicais. Vamos assumir agora que um nó não ramificado herda a extensão do nó imediatamente dominado por ele, como já fizemos no capítulo anterior.

Princípio dos nós não ramificados
Se α é um nó não ramificado cujo único constituinte imediato é β, então $[\![\alpha]\!] = [\![\beta]\!]$.

Assim, temos:

$[\![\text{NP}]\!] = [\![\text{N}]\!] = [\![\text{João}]\!] =$ joão

$[\![\text{VP}]\!] = [\![\text{V}]\!] = [\![\text{trabalha}]\!] = \lambda x.\ x$ trabalha

Resta-nos agora determinar como a extensão de S é obtida a partir das extensões de NP e VP. A extensão de NP é um indivíduo, enquanto a extensão de VP é uma função que leva indivíduos em valores de verdade. Como a extensão de S deve ser um valor de verdade, o que precisamos é de um princípio que derive a extensão de S, aplicando a extensão de VP à extensão do NP. Isto nos daria o seguinte resultado:

$[\![\text{S}]\!]$ = $(\lambda x.\ x$ trabalha)(joão)

$[\![\text{S}]\!]$ = 1 *sse* João trabalha

Isso é exatamente o que queríamos derivar. Um sistema como o esquematizado acima se estende diretamente a outros casos de sentenças formadas por um verbo intransitivo e um nome próprio na posição de sujeito. Por exemplo, a extensão do nome *Maria* é o indivíduo chamado Maria, a de *Pedro*, o indivíduo chamado Pedro e assim por diante. Chamemos o conjunto de todos os indivíduos de D_e, o domínio dos indivíduos (ou entidades). Podemos afirmar então que a extensão de todo nome próprio pertence a D_e. Com relação aos verbos intransitivos em geral, podemos estender o que acabamos de ver com o verbo *trabalhar* para outros verbos dessa classe. Por exemplo, assumiremos que a extensão do verbo *estudar* é uma função f que leva indivíduos a valores de verdade, tal que, para todo indivíduo x, $f(x)$ é igual a 1 se, e somente se, x estuda. Se chamarmos o conjunto dos valores de verdade — $\{0, 1\}$ — de D_t, temos que a extensão de todo verbo intransitivo é uma função de $\mathbf{D}_e$ em D_t.

$[\![\text{estuda}]\!]$ = $\lambda x_e.\ x$ estuda

Note que, na representação acima, escrevemos λx_e em vez de $\lambda x : x \in D_e$. É comum representar a informação sobre o domínio de uma função através de um subscrito junto à variável associada ao operador lambda, exatamente como acabamos de fazer. Veremos mais exemplos disso no decorrer do livro.

Podemos formular agora aquele que será o pilar de todo nosso sistema interpretativo, o princípio de composição conhecido como APLICAÇÃO FUNCIONAL.

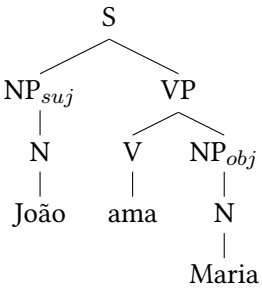
Aplicação funcional
Seja α um nó ramificado, cujos constituintes imediatos são β e γ. Se $[\![\beta]\!]$ é uma função e $[\![\gamma]\!]$ pertence ao domínio de $[\![\beta]\!]$, então $[\![\alpha]\!] = [\![\beta]\!]([\![\gamma]\!])$.

No exemplo mais acima, obtivemos a extensão da sentença *João trabalha* a partir das extensões de NP e VP, valendo-nos de aplicação funcional. Isso foi possível porque o domínio da função correspondente à extensão de VP era D_e, e a extensão do NP sujeito pertencia a esse domínio.

2.3 Verbos transitivos

Passemos, agora, a um exemplo com um verbo transitivo:

(2) João ama Maria.



A novidade aqui é o verbo *ama*, que se combina com o objeto direto *Maria*, formando um VP, e só depois com o sujeito *João*, formando a sentença. Queremos que nosso sistema interpretativo respeite a estrutura sintática da sentença e proceda composicionalmente, atribuindo uma extensão a cada nó dessa estrutura.

Vejamos como proceder. Sabemos o que queremos derivar:

$[\![\text{S}]\!]$ = 1 *sse* João ama Maria.

Já sabemos também que a extensão do NP sujeito é o indivíduo João. Concentremo-nos agora na extensão do VP *ama Maria*. Baseado no que vimos no exemplo anterior, obteremos o resultado desejado se assumirmos que a extensão de VP é uma função que leva indivíduos em valores de verdade. Neste caso, queremos uma função que leve um indivíduo x no valor de verdade 1 se, e somente se, x ama

Maria:

$[\![\text{VP}]\!] = \lambda x.\ x$ ama Maria

Passando para o interior de VP, já sabemos que a extensão do NP objeto é o indivíduo Maria. Nossa meta agora é descobrir uma extensão adequada para o verbo *amar*. Aqui podemos nos valer mais uma vez do princípio de aplicação funcional como guia. Se quisermos aplicá-lo neste caso, devemos assumir que a extensão de V, que é igual à extensão de *ama*, é uma função cujo domínio é D_e. Mas que tipo de objeto essa função deve retornar quando aplicada ao indivíduo Maria? Isso nós já sabemos: ela deve retornar a extensão de VP, ou seja, uma outra função:

$[\![\text{V}]\!]([\![\text{NP}_{obj}]\!]) = [\![\text{VP}]\!]$

Queremos, portanto, uma função que, aplicada a um indivíduo, retorne uma outra função. Não há nada de errado com isso. Funções podem tomar como argumentos e retornar como valores quaisquer tipos de objetos, inclusive outras funções. Funções assim são chamadas, por vezes, de FUNÇÕES DE ORDEM SUPERIOR. A notação lambda fornece uma maneira bastante compacta de representar essas funções. Para o verbo *ama*, temos:

$[\![\text{ama}]\!] = \lambda x_e.\lambda y_e.\ y$ ama x

Em palavras, a extensão de *ama* aplicada a um indivíduo *x* retorna uma outra função. Essa outra função leva um indivíduo *y* no valor de verdade 1 se e somente se *y* ama *x*, sendo esse *x* o argumento que alimentou a função original. Note que o argumento (x) dessa função original — a extensão de *ama* — será fornecido pela extensão do objeto direto da sentença, que, nesse caso, corresponde à pessoa amada. Já o argumento (y) da função resultante — a extensão de VP — será fornecido pela extensão do sujeito. Daí termos y ama x, e não x ama y na representação da função.

Para pôr em prática essas ferramentas que acabamos de adquirir, vamos calcular, passo a passo, a derivação das condições de verdade da sentença *Maria odeia João*, cuja estrutura é análoga à de *João ama Maria*, mas cujo significado obviamente não o é. As entradas lexicais que necessitamos estão dadas a seguir:

$[\![\text{Maria}]\!]$ = Maria

⟦João⟧ = João

⟦odeia⟧ = $\lambda x_e.\lambda y_e.$ y odeia x

Com as entradas lexicais acima mais o princípio dos nós não ramificados, obtemos a extensão do nó V e dos nós correspondentes aos NPs sujeito e objeto.

⟦V⟧ = ⟦odeia⟧

⟦[$_{\text{NP}}$ João]⟧ = ⟦[$_{\text{N}}$ João]⟧ = ⟦João⟧

⟦[$_{\text{NP}}$ Maria]⟧ = ⟦[$_{\text{N}}$ Maria]⟧ = ⟦Maria⟧

Já a aplicação funcional nos permite obter a extensão de VP a partir das extensões de V e do NP objeto:

⟦VP⟧ = ⟦V⟧(⟦[$_{\text{NP}}$ João]⟧)

⟦VP⟧ = ($\lambda x_e.\lambda y_e.$ y odeia x)(João)

⟦VP⟧ = $\lambda y_e.$ y odeia João (Conversão-λ)

Para obter as condições de verdade da sentença, utilizamos aplicação funcional mais uma vez:

⟦S⟧ = ⟦VP⟧(⟦[$_{\text{NP}}$ Maria]⟧)

⟦S⟧ = ($\lambda y_e.$ y odeia João)(Maria)

⟦S⟧ = 1 *sse* Maria odeia João

Podemos apreciar aqui um ponto vantajoso na utilização de funções para representar as extensões de predicados verbais. Apenas com aplicação funcional, fomos capazes de derivar as condições de verdade tanto de sentenças com verbos intransitivos quanto de sentenças com verbos transitivos. Conforme veremos mais adiante, outros tipos de predicados verbais e não verbais também receberão tratamento adequado via funções e aplicação funcional.

Antes de prosseguir, vamos introduzir uma notação que usaremos no restante deste livro e que, mais cedo ou mais tarde, o leitor que se aventurar pela literatura em semântica formal encontrará. Retomemos a extensão dos verbos *trabalhar* e *amar*, a título de exemplo:

$[\![\text{trabalhar}]\!] = \lambda x.\ x \text{ trabalha}$

$[\![\text{amar}]\!] = \lambda x.\lambda y.\ y \text{ ama } x$

É bastante comum nos textos da área utilizar, na parte da metalinguagem correspondente às especificações das condições de verdade, uma notação oriunda da lógica de predicados. Nela, sentenças simples formadas por um predicado (P) e seus argumentos ($a_1, a_2, ..., a_n$) são representadas por $P(a_1, a_2, ..., a_n)$. Nessa notação, o predicado aparece em primeiro lugar, seguido dos argumentos entre parênteses e separados por vírgulas. Assim, em vez de *x trabalha* ou *y ama x*, temos *trabalha(x)* ou *ama(y,x)*. Para diferenciar os predicados da linguagem objeto (no caso, o português) desses predicados da metalinguagem, é comum representar esses últimos por algum tipo especial de fonte tipográfica, tais como negrito, itálico, caixa alta ou versalete. Aqui, optaremos por essa última. Dessa forma, as extensões vistas acima para os verbos *trabalhar* e *amar* receberão as seguintes representações:

$[\![\text{trabalhar}]\!] = \lambda x.\ \textsc{trabalha}(x)$

$[\![\text{amar}]\!] = \lambda x.\lambda y.\ \textsc{ama}(y, x)$

Deve-se sempre ter em mente que essa é uma questão meramente notacional, adotada por conveniência e tradição na área. Do ponto de vista técnico, as funções acima são as mesmas funções com que trabalhamos anteriormente.

2.4 Tipos e domínios semânticos

Vimos acima que a extensão de um nome próprio pertence a D_e, o domínio dos indivíduos, e que a extensão de uma sentença pertence a D_t, o domínio dos valores de verdade. Vamos introduzir agora uma maneira de representar os domínios a que pertencem as extensões de verbos transitivos e intransitivos, além de outras variedades de constituintes que iremos discutir mais adiante.

Dizemos que a extensão de um constituinte sintático é de TIPO *e* se ela pertence

a D_e, e que a extensão de um constituinte é de tipo *t* se ela pertence a D_t. Para extensões funcionais, dizemos que a extensão de um constituinte é de tipo $\langle \sigma, \tau \rangle$ se ela é uma função que leva elementos de D_σ em D_τ. Nesse caso, $D_{\langle \sigma,\tau \rangle}$ é o conjunto de todas essas funções. A extensão de um verbo intransitivo é, então, de tipo $\langle e, t \rangle$, já que é uma função que leva indivíduos em valores de verdade, e $D_{\langle e,t \rangle}$ representa o domínio das funções de tipo $\langle e, t \rangle$. Já a extensão de um verbo transitivo é de tipo $\langle e, \langle e, t \rangle \rangle$, sendo $D_{\langle e,\langle e,t \rangle \rangle}$ o domínio das funções deste tipo. Como vimos, as extensões de verbos transitivos são funções que levam elementos de tipo *e* a funções de tipo $\langle e, t \rangle$.

Podemos definir recursivamente um número infinito de tipos semânticos, a partir dos tipos básicos *e* e *t*. Por definição recursiva, entenda-se uma definição que assume a existência de tipos básicos (no nosso caso, *e* e *t*) e define tipos complexos (no nosso caso, os tipos funcionais) a partir deles. Vejamos como:

Tipos semânticos

(i) *e* e *t* são tipos semânticos;

(ii) Se σ e τ são tipos semânticos, então $\langle \sigma, \tau \rangle$ é um tipo semântico;

(iii) Nada mais é um tipo semântico.

Pela cláusula (i), temos que tanto e quanto t são tipos semânticos. Sendo assim, de acordo com a cláusula (ii), temos que $\langle e, t \rangle$ também é um tipo semântico. Valendo-nos novamente da cláusula (ii), temos que $\langle e, \langle e, t \rangle \rangle$ também é um tipo semântico, já que tanto e quanto $\langle e, t \rangle$ o são. O mesmo vale para $\langle t, t \rangle$, $\langle \langle e, t \rangle, t \rangle$, e infinitos outros tipos funcionais, alguns dos quais aparecerão no decorrer deste livro. Notemos, por fim, que a cláusula (iii) afirma que apenas os objetos construídos a partir de (i) e (ii) são tipos semânticos.

Quanto aos domínios semânticos, eles são pareados aos tipos que acabamos de ver:

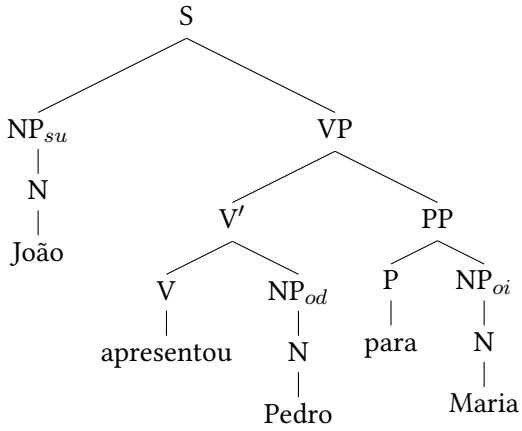
Domínios semânticos

D_e: conjunto dos indivíduos
D_t: conjunto dos valores de verdade ({0,1})
$D_{\langle \sigma,\tau \rangle}$: conjunto das funções de D_σ em D_τ.

Utilizaremos daqui em diante a notação acima para representar o tipo semântico (ou o domínio) a que pertence a extensão de um sintagma. Todos os itens lexicais e demais constituintes sintáticos que discutiremos no decorrer deste livro possuirão tipos semânticos que podem ser obtidos a partir da definição recursiva que acabou de ser apresentada.

2.5 Verbos bitransitivos

Resumindo o que vimos até aqui, podemos dizer que nosso sistema se baseia em um léxico, no qual estão especificadas as extensões de todos os itens lexicais e dois princípios de composição, que fornecem instruções sobre como obter a extensão de um constituinte sintático a partir de seus subconstituintes imediatos: O princípio de aplicação funcional e o princípio dos nós não ramificados. Vamos ver agora como esses princípios se aplicam na obtenção do significado de sentença com verbos bitransitivos, para os quais assumiremos a representação abaixo (mas ver exercício II ao final do capítulo).



Há dois itens lexicais nessa estrutura que ainda não discutimos: a preposição *para* e o verbo *apresentou*. Com relação à preposição, assumiremos tratar-se de um item semanticamente vácuo, inserido na estrutura por razões puramente sintáticas (caso, talvez). O sistema interpretativo desprezará esse item e tratará o nó PP como se fosse não ramificado. Fica faltando, então, uma extensão para o verbo *apresentou*. Vamos proceder de cima para baixo. O resultado final que desejamos é:

⟦S⟧ = 1 *sse* João apresentou Pedro para Maria.

Na notação importada da lógica de predicados, que começaremos a adotar a partir daqui, temos:

⟦S⟧ = 1 *sse* APRESENTOU(*joão,pedro,maria*)

Além de representar o predicado em versalete, representaremos os argumentos em itálico e minúsculas, demarcando melhor a distinção entre um nome próprio (João) e o indivíduo correspondente (*joão*). Dessa forma, com relação ao sujeito, a entrada lexical de *João* mais o princípio dos nós não ramificados nos fornecem o seguinte:

⟦NP_{su}⟧ = ⟦N⟧ = ⟦João⟧ = *joão*

Para que aplicação funcional possa nos servir, vamos assumir que VP seja uma função de tipo $\langle e, t\rangle$ que leva um indivíduo *x* no valor de verdade 1 se e somente se *x* apresentou Pedro para Maria.

⟦VP⟧ = λx_e. APRESENTOU(*x,pedro,maria*)

Quanto a PP, como este será tratado como um nó não ramificado, temos o seguinte:

⟦PP⟧ = ⟦NP_{oi}⟧ = ⟦N⟧ = ⟦Maria⟧ = *maria*

E quanto a V′? Novamente, guiados por aplicação funcional, podemos deduzir seu tipo semântico. Estamos buscando uma função que tomará a extensão de PP (tipo e) como argumento e retornará a extensão de VP (tipo $\langle e, t\rangle$), ou seja, uma função de tipo $\langle e, \langle e, t\rangle\rangle$.

⟦V′⟧ = λy_e. λx_e. APRESENTOU(*x,pedro,y*)

Com relação ao objeto direto, temos que:

⟦NP_{od}⟧ = ⟦N⟧ = ⟦Pedro⟧ = *pedro*

Por fim, chegamos ao verbo. Este deverá ser uma função que toma a extensão do

objeto direto (tipo e) como argumento e retorna a extensão de V′ (tipo $\langle e, \langle e, t\rangle\rangle$), ou seja, a extensão do verbo *apresentou* será uma função de tipo $\langle e, \langle e, \langle e, t\rangle\rangle\rangle$.

$[\![\text{apresentou}]\!] = \lambda z_e.\ \lambda y_e.\ \lambda x_e.\ \textsc{apresentou}(x,z,y)$

A extensão do verbo *apresentou* pode ser vista como uma função que toma três indivíduos *z, y, x* — um de cada vez e nessa ordem — como argumentos e retorna o valor de verdade 1 se, e somente se, *x* apresentou *z* pra *y*. Outros verbos bitransitivos como *dar* e *mostrar*, por exemplo, podem ser tratados da mesma maneira.

É crucial notar que o primeiro argumento (z) na representação acima está vinculado, por assim dizer, com a posição de objeto direto, o segundo (y) com a posição de objeto indireto e o terceiro (x) com a posição de sujeito. Com isso, queremos chamar a atenção ao fato de que a escolha da letra (x, y, ou z) para representar uma posição argumental não é, em si mesma, relevante. O que realmente importa é essa vinculação com as devidas posições. Dessa forma, poderíamos igualmente representar a extensão do verbo *apresentar* das seguintes formas:

$[\![\text{apresentou}]\!] = \lambda x.\ \lambda y.\ \lambda z.\ z$ apresentou x para y

$[\![\text{apresentou}]\!] = \lambda x.\ \lambda y.\ \lambda z.\ \textsc{apresentou}(z,x,y)$

Trata-se da mesma função anterior, já que aqui também temos o primeiro argumento (x) vinculado à posição de objeto direto, o segundo (y) à de objeto indireto, e o terceiro (z) à de sujeito (a esse respeito, ver o exercício I ao final do capítulo).

2.6 Predicados não verbais

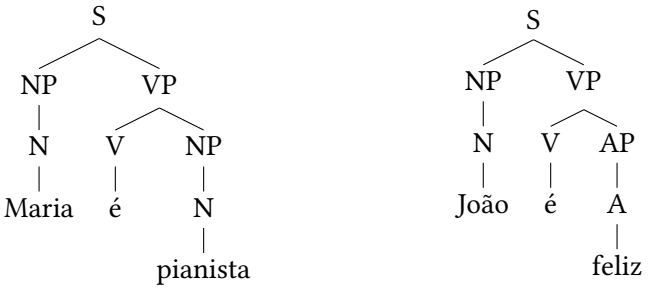
Nosso próximo passo será a análise de sentenças contendo predicados não verbais, como substantivos comuns, adjetivos e algumas preposições, como em (3) abaixo:

(3) a. Maria é pianista.
    b. João é feliz.
    c. Pedro está em Campinas.

Como não discutiremos temas ligados a tempo e aspecto neste curso, passaremos por cima da distinção entre *ser* e *estar*, e assumiremos, apenas por conveniência, que ambos são sinônimos, sem maiores justificativas. Isso é obviamente uma

simplificação, mas uma discussão mais aprofundada nos afastaria dos temas mais centrais deste livro. Iremos nos referir tanto a *ser* quanto a *estar* através de um termo mais genérico: cópula.

Vamos assumir para sentenças como (3a) e (3b) as estruturas abaixo:



Substantivos comuns e adjetivos como os acima guardam certa semelhança com verbos intransitivos. Assim como alguém que sabe o significado do verbo *fumar* sabe as condições para que se possa dizer de um indivíduo qualquer se ele fuma ou não, uma pessoa que sabe o significado do substantivo comum *pianista* ou do adjetivo *feliz* sabe as condições para que se possa dizer de uma pessoa qualquer se ela é ou não pianista ou se ela é feliz ou não. Dada essa semelhança, iremos propor para esses predicados extensões de tipo $\langle e, t\rangle$:

$[\![\text{pianista}]\!] = \lambda x.\ x$ é pianista

$[\![\text{feliz}]\!] = \lambda x.\ x$ é feliz

Na notação de predicados, costuma-se omitir o verbo *ser*:

$[\![\text{pianista}]\!] = \lambda x.\ \textsc{pianista}(x)$

$[\![\text{feliz}]\!] = \lambda x.\ \textsc{feliz}(x)$

Assumiremos que a cópula é um elemento semanticamente vácuo, e que, portanto, a extensão de VP nos exemplos acima é a mesma de AP e NP (mas ver exercício V ao final do capítulo):

$[\![\text{VP}]\!] = [\![\text{NP}]\!] = \lambda x_e.\ \textsc{pianista}(x)$

$[\![\text{VP}]\!] = [\![\text{AP}]\!] = [\![\text{feliz}]\!] = \lambda x_e.\ \textsc{feliz}(x)$

Com isso, podemos nos valer de aplicação funcional para obter as condições de verdade destas sentenças a partir das extensões dos NPs sujeito e dos respectivos VPs. Para (3a), temos:

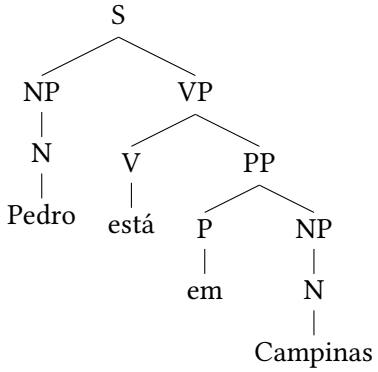
$[\![\text{S}]\!] = [\![\text{VP}]\!]([\![\text{NP}]\!])$

$[\![\text{S}]\!] = (\lambda x_e.\ \textsc{pianista}(x))(\textit{joão})$

$[\![\text{S}]\!] = 1$ *sse* $\textsc{pianista}(\textit{joão})$

Para (3b), procedemos de maneira inteiramente análoga.

Passemos agora a (3c), que envolve um sintagma preposicional:



O paralelo neste caso é entre a preposição *em* e verbos transitivos. Intuitivamente, ambos relacionam dois indivíduos. Assim como alguém que sabe o significado do verbo transitivo *amar* sabe as condições para que se possa dizer de dois indivíduos x e y se x ama y ou não, uma pessoa que saiba o significado da preposição locativa *em* sabe as condições para que se possa dizer de dois indivíduos x e y se x está ou não em y. A entrada lexical que proporemos para essa preposição faz jus a esse paralelo, tratando a extensão de *em* como uma função de tipo $\langle e, \langle e, t\rangle\rangle$.

$[\![\text{em}]\!] = \lambda x_e.\ \lambda y_e.\ \textsc{em}(y,x)$

O cálculo das condições de verdade de (3c) procede de maneira análoga aos casos envolvendo verbos transitivos. Para a obter a extensão de PP, utilizamos aplica-

ção funcional:

$[\![\text{PP}]\!] = [\![\text{P}]\!]([\![\text{NP}]\!])$

$[\![\text{PP}]\!] = (\lambda x.\ \lambda y.\ \textsc{em}(y,x))(\textit{campinas})$

$[\![\text{PP}]\!] = \lambda y.\ \textsc{em}(y,\textit{campinas})$

Como a cópula é um item semanticamente vácuo, temos que a extensão de VP é igual à extensão de PP. Por fim, utilizamos aplicação funcional mais uma vez para obtermos as condições de verdade da sentença:

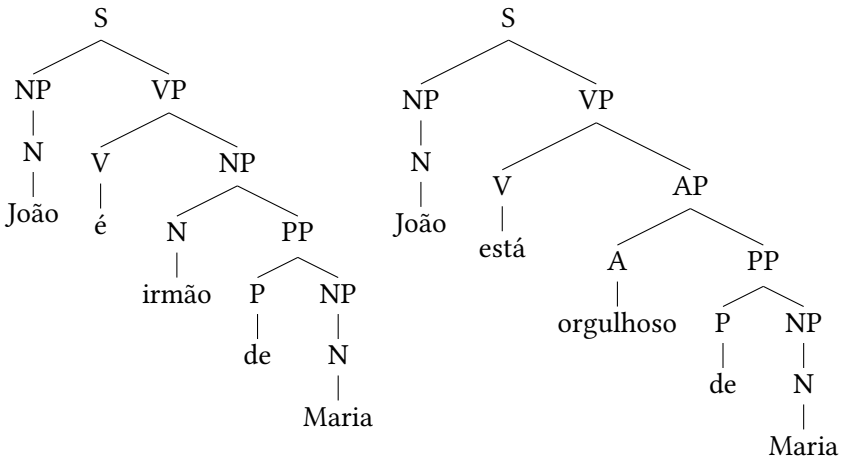
$[\![\text{S}]\!] = [\![\text{VP}]\!]([\![\text{NP}]\!])$

$[\![\text{S}]\!] = (\lambda y.\ \textsc{em}(y,\textit{campinas}))(\textit{pedro})$

$[\![\text{S}]\!] = 1$ *sse* $\textsc{em}(\textit{pedro},\textit{campinas})$

Além de adjetivos e nomes comuns como *feliz* e *pianista*, que poderíamos chamar de intransitivos, há também casos que se assemelham a verbos transitivos, como, por exemplo, *orgulhoso* e *irmão*:

(4) a. João está orgulhoso de Maria.
b. João é irmão de Maria.



Para esses casos, vamos assumir que a preposição *de* é semanticamente vácua. Isso permitirá que a extensão do NP complemento de P nas estruturas acima possa servir de argumento para a extensão de *orgulhoso* e *irmão*, que trataremos como sendo funções de tipo $\langle e, \langle e, t\rangle\rangle$, a exemplo de outros predicados transitivos que já discutimos.

$[\![\text{orgulhoso}]\!] = \lambda x.\ \lambda y.\ \textsc{orgulhoso}(y,x)$

$[\![\text{irmão}]\!] = \lambda x.\ \lambda y.\ \textsc{irmão}(y,x)$

Fica como exercício para o leitor o cálculo, passo a passo, das condições de verdade de (4a) e (4b).

2.7 Relações, funções e *currying*

Antes de fechar o capítulo, mais uma seção de cunho técnico, mas importante para o leitor se familiarizar com certas práticas e jargões da área.

Notamos na seção inicial deste capítulo que há uma correspondência biunívoca entre conjuntos e funções características construídos a partir de certo domínio. Nos casos que analisamos até aqui, o domínio relevante é D_e, o domínio dos indivíduos. No caso dos verbos intransitivos, notamos correspondências como as abaixo, em que a é um indivíduo qualquer:

$[\![\text{trabalha}]\!](a) = 1 \leftrightarrow a \in \{x \mid x \text{ trabalha}\}$

$[\![\text{trabalha}]\!](a) = 0 \leftrightarrow a \notin \{x \mid x \text{ trabalha}\}$

Tais equivalências se estendem para os casos de verbos transitivos e bitransitivos. Entretanto, em vez de caracterizar conjuntos de indivíduos, as extensões dos verbos transitivos e bitransitivos caracterizam RELAÇÕES entre indivíduos. Relações nada mais são que conjuntos de sequências ordenadas, também chamadas de TUPLAS. Assim, verbos transitivos são frequentemente apresentados como denotando conjuntos de pares ordenados (relações binárias) e os bitransitivos como denotando conjuntos de ternos ordenados (relações ternárias), como ilustrado abaixo para os verbos *amar* e *apresentar*, respectivamente:

$R_{amar} = \{\langle x, y\rangle | x \text{ ama } y\}$

$R_{apresentar}$ = $\{\langle x, y, z\rangle | x$ apresenta y para $z\}$

Os colchetes angulados ($\langle\rangle$) indicam tratar-se de uma sequência ordenada e que, portanto, $\langle x, y\rangle$ será diferente de $\langle y, x\rangle$ sempre que x for diferente de y. No caso de *amar*, por exemplo, um par $\langle a, b\rangle$ pertencerá à relação sempre que o primeiro membro do par (a) amar o segundo (b).

Novamente, essa mudança de funções para relações é inofensiva, se atentarmos para as seguintes equivalências, em que a, b e c são três indivíduos quaisquer:

$[\![\text{amar}]\!](a)(b) = 1 \leftrightarrow \langle b, a\rangle \in R_{amar}$

$[\![\text{apresentar}]\!](a)(b)(c) = 1 \leftrightarrow \langle c, a, b\rangle \in R_{apresentar}$

Há ainda um ponto relacionado ao que acabamos de ver e que é importante o leitor conhecer. Primeiramente, notemos que, a partir da função correspondente à extensão de um verbo transitivo, para chegarmos a um valor de verdade, procedemos em dois passos. Inicialmente, aplicamos a função a um indivíduo, obtendo uma outra função. Essa outra função, por sua vez, quando aplicada também a um indivíduo, retorna um valor de verdade (0 ou 1).

$[\![\text{amar}]\!](x)(y) = 1$ *sse* y ama x

O mesmo se deu com verbos bitransitivos, sendo que, nesse caso, o processo envolveu três passos.

$[\![\text{apresentar}]\!](x)(y)(z) = 1$ *sse* z apresenta x para y

Não fosse nosso intuito de fazer a derivação semântica corresponder à estrutura sintática binária que estamos assumindo, poderíamos reduzir esses processos a um único passo, identificando as extensões verbais com funções que tomam sequências (tuplas) de indivíduos retornando diretamente um valor de verdade. Dessa forma, em vez de denotações como as em (a) nos exemplos abaixo, teríamos aquelas em (b):

(5) a. $[\![\text{ama}_1]\!] = \lambda x.\lambda y.\ y$ ama x
b. $[\![\text{ama}_2]\!] = \lambda\langle x, y\rangle.\ y$ ama x

(6) a. $[\![\text{apresenta}_1]\!] = \lambda x.\lambda y.\lambda z.\ z$ apresenta x pra y
b. $[\![\text{apresenta}_2]\!] = \lambda\langle x, y, z\rangle.\ z$ apresenta x pra y

No caso de *amar*, o argumento da função seria um par ordenado, enquanto no caso de *apresentar*, teríamos um terno ordenado. Matematicamente, essa mudança é inofensiva se atentarmos para as seguintes igualdades, válidas para quaisquer x, y, z:

$[\![\text{ama}_1]\!](x)(y) = [\![\text{ama}_2]\!](\langle x, y\rangle)$

$[\![\text{apresenta}_1]\!](x)(y)(z) = [\![\text{apresenta}_2]\!](\langle x, y, z\rangle)$

Essas igualdades se generalizam para funções com qualquer número de argumentos. A transformação de uma função f qualquer que toma uma sequência $\langle a_1, a_2, ..., a_n\rangle$ como argumento em uma outra função f' que toma *n* argumentos, $a_1, a_2, ..., a_n$, um de cada vez, obedecendo à equivalência acima é chamada de *curryamento (currying)* ou *shönfinkelização*, em homenagem a Haskell Curry e Moses Schönfinkel, pioneiros na descrição e aplicação desse processo. A importância de conhecer essas representações e essas equivalências é que a literatura da área está cheia de alternâncias entre elas e os seus respectivos jargões.

Apêndice: mais sobre conversões lambda e alfa

Apresentamos no início deste capítulo a notação lambda para a representação de funções e os processos de conversão-λ e conversão-α. Vejamos agora alguns cuidados necessários na manipulação desses processos. Comecemos pela função abaixo:

$\lambda n.\ 5$

Note que se trata de uma função um tanto peculiar, já que não há qualquer ocorrência da variável *n* no corpo da função. Isso significa que o valor retornado pela função não depende da natureza do argumento a que ela se aplica. Qualquer que seja esse argumento, o valor retornado será sempre o número 5. Uma função como essa é chamada de FUNÇÃO CONSTANTE. Funções de ordem superior também podem ser constantes, como deixam claro os exemplos abaixo:

$\lambda n.(\lambda m.\ m + 1)$

$\lambda f.\ 3$

No primeiro caso, não importa o argumento: o valor retornado é sempre a função sucessor. No segundo caso, qualquer que seja a função que sirva de argumento, o valor retornado é sempre o número 3.

Considere agora o exemplo abaixo:

$\lambda n.(\lambda n.\ n + 1)$

Note que o corpo dessa função é a função sucessor. Se formos descrever a função acima em palavras, diríamos tratar-se de uma função que toma números naturais como argumentos e que retorna a função sucessor como valor. Estamos diante de uma função constante! Alguém poderia perguntar: mas a expressão $n + 1$ não contém uma ocorrência da variável *n*? Sim, mas trata-se de uma ocorrência *ligada* por um operador lambda no próprio corpo da função. A conclusão que podemos tirar disso é que uma expressão do tipo $\lambda v.\alpha$ representará uma função constante sempre que a variável *v* não aparecer *livre* em α.

A aplicação da função acima a um número qualquer — 3, por exemplo — pode ser representada e simplificada por conversão-λ da seguinte forma:

$(\lambda n.\ \lambda n.n + 1)(3)$
$(\lambda n.\ n + 1)$ (conversão-λ)

Com isso queremos deixar claro que, no processo de conversão-λ aplicado a uma expressão do tipo $(\lambda v.\gamma)(a)$, apenas as *ocorrências livres* de *v* em γ devem ser substituídas por *a*.

Ainda em conexão com o que estamos vendo, compare as duas funções abaixo:

(7) $(\lambda n.(\lambda m.\ m + n))$

(8) $(\lambda m.(\lambda m.\ m + m))$

Note que se trata de duas funções distintas. Isso fica claro ao percebermos que a segunda, mas não a primeira, é uma função constante. A questão que se coloca aqui é a seguinte: se olharmos atentamente, veremos que a função em (8) é o resultado da aplicação cega do processo de conversão-α à função em (7), ou seja, da substituição de todas as ocorrências de n por m. Mas, quando introduzimos esse processo, salientamos que sua aplicação à representação de uma função preserva a identidade dessa função. O que então ocorreu de errado no caso acima? Note que em (7) a variável *n* ocorria livre no corpo da função. Ao ser substituída por *m* na expressão $m+n$, a nova ocorrência de *m* acabou *acidentalmente ligada* por um

outro operador lambda no corpo da nova função, que acabou se transformando em uma função constante.

Para que o processo de conversão-α preserve sempre a identidade de uma função, sua aplicação só é considerada lícita se a variável que está sendo substituída não acabar acidentalmente ligada no corpo da nova função.

Assim como a conversão-α, a conversão-λ também tem seus caprichos. Antes de discuti-los, vamos fazer algumas considerações preliminares. Se alguém nos pergunta a que a expressão 2 + 3 se refere, não temos dúvida em responder que se refere ao número 5. Se, no entanto, nos é perguntado a que a expressão 2 + n se refere, ficamos sem uma resposta direta. A razão é óbvia: essa expressão contém uma ocorrência livre de uma variável e, a menos que nos seja dito que valor atribuir a essa variável, não é possível identificar a que a expressão se refere.

Expressões mais complexas envolvendo funções e o operador lambda estão sujeitas ao mesmo tipo de comentário. Considere, por exemplo, as duas funções abaixo:

(9) $\lambda n.\ n + 1$

(10) $\lambda n.\ n + m$

No caso de (9), está claro diante de que função nós estamos: a função sucessor. Já no caso de (10), a menos que nos seja informado que valor atribuir a variável *m*, ficamos sem saber com que função estamos lidando. Novamente, a razão para isto está no fato de que a variável *m* aparece livre na expressão em questão. Variáveis livres podem aparecer não apenas no interior de uma abstração, mas também na expressão do argumento aplicado a uma função. Compare os dois casos a seguir:

(11) $(\lambda n.\ n + 1)(3)$

(12) $(\lambda n.\ n + 1)(m)$

Em (11), temos a aplicação da função sucessor ao número 3. Não há variáveis livres e a expressão deve ter um valor definido. Ao simplificar essa expressão através da conversão-λ isso fica claro:

$(\lambda n.\ n + 1)(3)$
3 + 1
4

No caso de (12), a expressão contém uma variável livre e, portanto, a ela não podemos atribuir um valor definido. Novamente, a conversão-λ deixa isso claro:

$(\lambda n.\ n + 1)(m)$
$m + 1$

Note a coerência nos resultados obtidos. A expressão que não continha variáveis livres antes da conversão-λ (e que, portanto, tinha um valor definido) foi convertida em uma outra expressão também sem variáveis livres, continuando a ter um valor definido. Já a expressão que continha uma variável livre antes da conversão-λ foi convertida em uma expressão que também contém uma variável livre, continuando portanto sem um valor definido. Esta é uma característica da conversão-λ: expressões são simplificadas, mas sem alterar sua interpretação. Assim, não desejamos que uma aplicação do processo de conversão-λ elimine variáveis livres ou as introduza onde antes elas não ocorriam.

Com isso em mente, considere a seguinte simplificação:

(13) $(\lambda m.\ \lambda n.\ m + n)(n)$
$\lambda n.\ n + n$

Na primeira linha, temos a aplicação de uma função a um argumento, que, neste caso, é uma ocorrência da variável *n*. Essa ocorrência aparece livre na expressão, a qual, portanto, não tem um valor definido. A segunda linha é o resultado da conversão-λ. Seguimos a receita que já sabemos de cor: substituímos todas as ocorrências livres da variável *m* no corpo da função pelo argumento em questão, no caso a variável *n*. Entretanto, o resultado foi uma expressão sem variáveis livres e, portanto, com um valor definido. De fato, trata-se da função que leva um número natural *n* qualquer no seu dobro *2n*. Onde está o problema? Note que, ao substituir a variável *m* por *n*, essa última acabou acidentalmente ligada pelo operador lambda no corpo da função. É exatamente isso que precisamos proibir. E não é só em casos em que o argumento da função é uma variável que esse problema pode ocorrer. Casos em que o argumento contém uma ocorrência livre de uma variável também são suscetíveis ao problema, conforme mostra o exemplo abaixo:

(14) $(\lambda m.\lambda n.m + n)(n + 2)$
$\lambda n.n + n + 2$
$\lambda n.2n + 2$

Novamente, a conversão-λ nos levou de uma expressão contendo uma ocorrência livre de uma variável para outra expressão sem variáveis livres. Para evitar essa consequência indesejada, só se considera lícito o uso da conversão-λ em uma expressão formada por uma função $\lambda v.\gamma$ aplicada a um argumento *a* que obedecer à seguinte condição: se *x* é uma ocorrência livre de uma variável em *a*, então a substituição das ocorrências livres de *v* pelo argumento *a* no corpo da função não pode resultar na ligação de *x* no corpo dessa função.

Como, então, simplificar a aplicação funcional em (13) via conversão-λ? A saída é usar a conversão-α antes de usar a conversão-λ. Nesse caso, a conversão-α se aplicará à subexpressão que aparece sublinhada abaixo, substituindo-se *n* por uma variável que ainda não apareça na expressão, como *k*, por exemplo:

(15) $(\lambda m.\ \underline{\lambda n.\ m + n})(n)$
$(\lambda m.\ \lambda k.\ m + k)(n)$ Conversão-α
$\lambda k.\ n + k$ Conversão-λ

Essa é, de fato, a grande utilidade da conversão-α: preparar uma expressão para a aplicação da conversão-λ. Finalizamos aqui nossa discussão sobre esses dois processos, complementando a breve introdução à notação lambda para a representação de funções apresentada no início do capítulo.

Sugestões de leitura

A apresentação da notação lambda neste capítulo pode ser qualificada de instrumental, já que o objetivo foi utilizar tal ferramenta na construção do sistema interpretativo que começamos a desenvolver. Os leitores interessados em um tratamento formal mais rigoroso, no qual a notação lambda é integrada a um dos sistemas lógicos conhecidos como *cálculo lambda* devem consultar a excelente introdução em Carpenter (1997), capítulo 2. Assumimos neste capítulo que predicados verbais têm como extensões funções que tomam indivíduos como argumentos. Para um tratamento diferente, baseado na ideia de que (alguns) predicados verbais tomam eventos como argumentos, ver Davidson (1967). Para extensões das ideias de Davidson, incorporando a noção de papel temático nas relações entre predicados e argumentos, ver Parsons (1990) e Schein (1993). Ver também Kratzer (2003) para um tratamento composicional.

Exercícios

I. Na seção sobre verbos bitransitivos, dissemos que as extensões abaixo eram equivalentes:

⟦apresentou⟧ = $\lambda z.\ \lambda y.\ \lambda x.\ x$ apresentou z para y

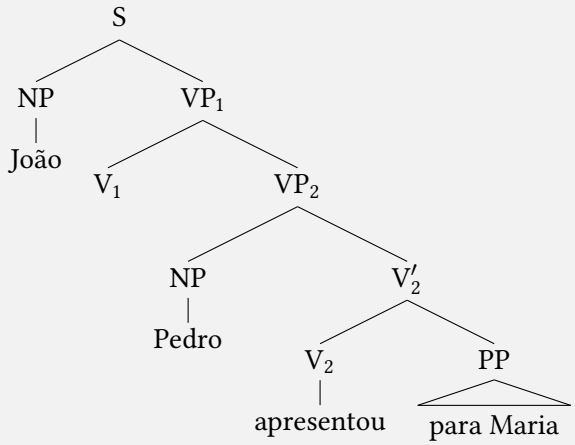
⟦apresentou⟧ = $\lambda x.\ \lambda y.\ \lambda z.\ z$ apresentou x para y

Calcule, passo a passo, as condições de verdade de *João apresentou Pedro para Maria* usando cada uma delas e confira se os resultados são os mesmos. Em seguida, faça o mesmo com a seguinte extensão:

⟦apresentou⟧ = $\lambda x.\ \lambda y.\ \lambda z.\ x$ apresentou y para z

O resultado continua o mesmo? Por quê?

II. Já se propôs na literatura sintática que a estrutura argumental de verbos bitransitivos como *apresentar* é, na verdade, algo como o seguinte (ver Carnie (2013), capítulo 14, para discussão e referências):



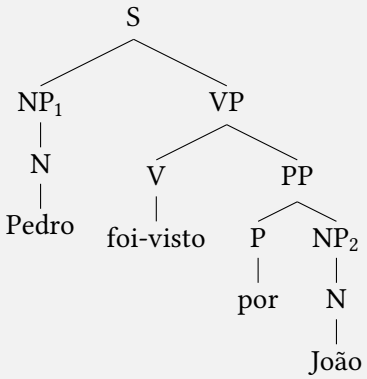
Nesse caso, uma operação sintática (sem reflexos semânticos) deslocaria o verbo para a posição V_1, logo após o sujeito, originando a ordem superficial S V OD OI. Assumindo que é a estrutura acima (sem movimento)

que chega ao componente semântico, e que V_1 seja semanticamente vácuo, proponha uma entrada lexical para o verbo apresentar, prestando bastante atenção no que foi visto no exercício anterior.

III. Considere o par de sentenças abaixo:

(a) João viu Pedro.
(b) Pedro foi visto por João.

Para (b), assuma a seguinte estrutura:

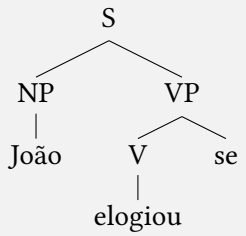


Para efeitos deste exercício, assuma que tanto *viu* quanto *foi visto* são itens lexicais e que a preposição *por* é semanticamente vácua. Proponha então uma extensão para *viu* e outra para *foi-visto* que dê conta do fato de que as duas sentenças acima têm as mesmas condições de verdade. Qual a relação entre essas duas extensões que você acabou de propor?

IV. Considere a sentença abaixo:

(a) João elogiou-se.

Assuma para essa sentença a seguinte estrutura sintática:



Assuma agora que o pronome reflexivo *se* tenha a seguinte extensão:

$[\![\text{se}]\!] = \lambda F.\lambda x.F(x)(x) = 1$

(i) Calcule as condições de verdade de (a) valendo-se da entrada lexical acima e verifique se o resultado obtido é ou não adequado.

(ii) Qual o tipo semântico da extensão acima? Qual o tipo do argumento que essa extensão toma? Qual o tipo do valor que ela retorna? Essa extensão costuma ser chamada de função de reflexivização ou intransitivização. Pense no tipo de transformação que ela efetua sobre seu argumento e diga o porquê de ela ser chamada por esses nomes.

V. Neste capítulo, assumimos que o verbo de ligação (cópula) em uma sentença como (a) é semanticamente vácuo:

(a) João é pianista

Entretanto, é possível atribuir a esse verbo uma extensão, chamada de função identidade, que pode ser representada como em (b) ou como em (c) abaixo:

(b) $[\![\text{é}]\!] = \lambda f_{\langle e,t\rangle}.\ f$
(c) $[\![\text{é}]\!] = \lambda f_{\langle e,t\rangle}.\lambda x_e.\ f(x) = 1$

(i) Mostre que tanto (b) quanto (c) derivam condições de verdade adequadas para (a).

(ii) Mostre que (b) e (c) são a mesma função.

3 Coordenação e negação

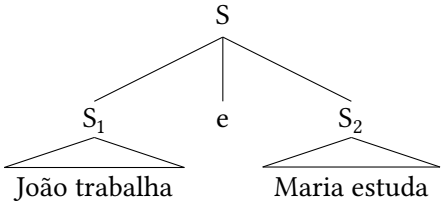
Neste capítulo, discutiremos o significado flexível da conjunção aditiva *e* e da palavra negativa *não*. Desenvolveremos uma análise em que as extensões desses itens são uma família de funções, continuando a nos valer de aplicação funcional como o pilar de nosso sistema composicional.

3.1 Coordenação

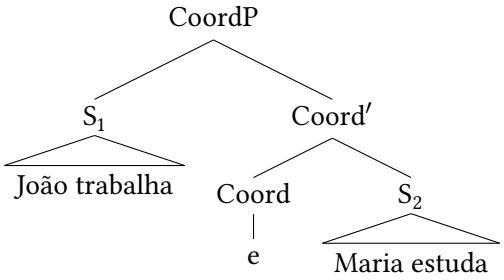
Consideremos a sentença (1) abaixo:

(1) João trabalha e Maria estuda.

Essa sentença é formada pela coordenação de duas outras sentenças por meio da conjunção aditiva *e*. No capítulo 1, representamos essa coordenação como uma estrutura ternária em que a conjunção *e* tinha como irmãos na árvore sintática as duas sentenças coordenadas.



É possível, entretanto, adotar uma estrutura binária para essas construções, colocando-as em pé igualdade com outras construções sintáticas complexas como os VPs e as sentenças do capítulo 2. Não iremos aqui argumentar empiricamente a favor dessa posição (ver, por exemplo, Munn (1993), capítulo 2, para possíveis indícios), apenas adotando-a por uma questão de uniformidade. Rotularemos o item *e* de Coord(enação) e as projeções imediatamente acima dele de Coord′ e CoordP:



Lembre-se, entretanto, que rótulos não desempenham nenhum papel em nosso sistema. O que realmente importa é a estrutura hierárquica e as extensões dos seus constituintes. Nem aplicação funcional nem o princípio dos nós não ramificados fazem menção a rótulos categoriais (NP, VP, S, ...). Portanto, se, por alguma razão extra, você preferir outros rótulos (ConjP, S′, ...), sinta-se à vontade para modificá-los.

Note a semelhança dessa estrutura com a estrutura de sentenças com verbos transitivos que discutimos no capítulo anterior. Tanto a conjunção *e* quanto um verbo transitivo aparecem junto a dois constituintes da mesma categoria: S no primeiro caso, e NP no segundo (o NP sujeito e o NP objeto). A diferença é que a extensão de uma sentença é de tipo t, enquanto a extensão de um NP é de tipo e. Tratamos os verbos transitivos como funções de tipo $\langle e, \langle e, t\rangle\rangle$. Iremos agora propor uma entrada lexical para a conjunção *e* que a trata como uma função de tipo $\langle t, \langle t, t\rangle\rangle$. A mudança nos tipos reflete a diferença que acabamos de notar. Quanto à descrição dessa função, notemos que uma sentença coordenada é verdadeira se as sentenças que a compõem forem ambas verdadeiras, e falsa se umas delas, ou ambas, forem falsas. A entrada lexical abaixo capta essa condição:

$[\![\text{e}]\!] = \lambda p_t.\ \lambda q_t.\ p = 1 \text{ e } q = 1$

Antes de prosseguir, pausemos um instante para fazer algumas considerações sobre o uso da conjunção *e*, já que esse uso costuma trazer à tona outras nuances de significado, além da adição, tais como nexos causais e temporais. Considere, por exemplo, as sentenças abaixo:

(2) João faltou ao serviço e seu patrão o demitiu.

(3) Pedro entrou na sala e João o cumprimentou.

O uso de (2) comumente veicula a ideia de que a falta ao serviço causou a demissão. Já o de (3) passa a ideia que primeiro Pedro entrou na sala, e depois João o

cumprimentou. Entretanto, esses aspectos do significado não parecem estar embutidos na contribuição semântica de *e*, já que podem facilmente ser cancelados:

(4) João faltou ao serviço e seu patrão o demitiu, mas uma coisa não teve nada a ver com a outra.

(5) Pedro entrou na sala e João o cumprimentou, não necessariamente nessa ordem.

Já a exigência de que ambas as orações coordenadas sejam verdadeiras tem um estatuto diferente e parece inexoravelmente atrelada a essa conjunção. Note a flagrante contradição nos casos abaixo:

(6) João faltou ao serviço e seu patrão o demitiu, mas João não faltou ao serviço.

(7) Pedro entrou na sala e João o cumprimentou, mas João não o cumprimentou.

Não discutiremos aqui como a veiculação de nexos causais e temporais emerge de certos usos da conjunção *e*. Limitamo-nos a apontar que esse é, foi e continua sendo um tema muito discutido em pragmática, que, muitas vezes, se caracteriza por buscar manter uma semântica mais austera, enriquecida por princípios conversacionais que regem a interação linguística entre falantes (ver sugestões de leitura ao final do capítulo).

Para realçar esse tratamento semântico enxuto que estamos dando a *e*, vamos usar na metalinguagem o conectivo lógico & (às vezes também representado por $\wedge$), caracterizado por exigir única e exclusivamente a verdade de ambas as sentenças coordenadas. Sua contribuição semântica costuma ser representada em uma tabela como a abaixo, em que ϕ e ψ representam duas sentenças quaisquer:

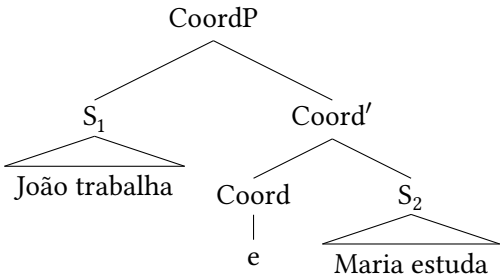
(8) Tabela de verdade de &

ϕ	ψ	ϕ & ψ
1	1	1
1	0	0
0	1	0
0	0	0

A entrada lexical de *e* será, portanto, a seguinte:

$[\![\text{e}]\!] = \lambda p_t.\ \lambda q_t.\ p = 1\ \&\ q = 1$

Feitas essas considerações, vejamos como as condições de verdade de (1), repetida abaixo, são derivadas



Através de aplicação funcional, obtemos o seguinte:

$[\![\text{Coord}']\!] = [\![\text{Coord}]\!]([\![\text{S}_2]\!])$

$= (\lambda p_t.\ \lambda q_t.\ p = 1\ \&\ q = 1)([\![\text{S}_2]\!])$
$= \lambda q_t.\ [\![\text{S}_2]\!] = 1\ \&\ q = 1$

Como já sabemos que:

$[\![\text{S}_2]\!] = 1$ *sse* ESTUDA(*maria*)

Temos que:

$[\![\text{Coord}']\!] = \lambda q_t.$ ESTUDA(*maria*) $\&\ q = 1$

Aplicação funcional é então novamente utilizada para derivar as condições de verdade de CoordP:

$[\![\text{CoordP}]\!] = [\![\text{Coord}']\!]([\![\text{S}_1]\!])$

$=$ ($\lambda q_t.$ ESTUDA(*maria*) $\&\ q = 1$)($[\![\text{S}_1]\!]$)
$= 1$ *sse* ESTUDA(*maria*) $\&\ [\![\text{S}_1]\!] = 1$

Como já sabemos que:

$[\![S_1]\!]$ = 1 *sse* TRABALHA(*joão*)

temos que

$[\![CoordP]\!]$ = 1 *sse* ESTUDA(*maria*) & TRABALHA(*joão*)

Note que, como o conectivo & utilizado na metalinguagem é simétrico, ou seja, a ordem dos elementos coordenados não é relevante, podemos igualmente escrever:

$[\![CoordP]\!]$ = 1 *sse* TRABALHA(*joão*) & ESTUDA(*maria*)

Esse é, de fato, o resultado que desejávamos.

3.2 A flexibilidade da coordenação

Analisamos acima uma sentença formada pela coordenação de duas outras sentenças, e atribuímos uma entrada lexical para a conjunção *e* de tipo $\langle t, \langle t, t\rangle\rangle$, que nos forneceu condições de verdade adequadas.

A conjunção *e*, entretanto, é capaz de coordenar não só sentenças mas uma variedade de constituintes, conforme atestam os exemplos abaixo:

(9)	a.	João estuda e trabalha.	Coord. de Vs intransitivos
	b.	João abraçou e beijou Maria.	Coord. de Vs transitivos
	c.	João mostrou e deu o livro pra Maria.	Coord. de Vs bitrans.
	d.	João abraçou Maria e beijou Paula.	Coord. de VPs
	e.	João é médico e psicólogo.	Coord. de NPs
	f.	João está alegre e orgulhoso da Maria.	Coord. de APs
	g.	João esteve em Paris e em Londres.	Coord. de PPs.
	h.	João e Maria estudam.	Coord. de nomes próprios

Como se pode notar, todos os tipos de constituintes que vimos até agora podem ser coordenados pela da conjunção *e*. Porém, nosso sistema não é capaz de interpretar nenhuma das sentenças acima. A razão para isso é simples. A entrada lexical que fornecemos para a conjunção *e*, por ser de tipo $\langle t, \langle t, t\rangle\rangle$, só permite sua combinação com constituintes de tipo *t*, ou seja, constituintes sentenciais. Mas nenhum dos constituintes coordenados em (9a)-(9h) é de tipo *t*. Os verbos

intransitivos em (9a), por exemplo, denotam funções de tipo $\langle e, t\rangle$, assim como os VPs em (9d) e os predicados não verbais em (9e)-(9g). Já os verbos transitivos em (9b) denotam funções de tipo $\langle e, \langle e, t\rangle\rangle$, enquanto os bitransitivos em (9c) denotam funções de tipo $\langle e, \langle e, \langle e, t\rangle\rangle\rangle$. Por fim, os nomes próprios em (9h) denotam indivíduos (tipo e).

Uma saída para esse problema seria admitir que as sentenças em (9) são, na verdade, versões reduzidas de sentenças mais complexas envolvendo coordenação sentencial. De fato, para todas as sentenças em (9), é possível encontrar uma sentença equivalente envolvendo a coordenação de duas outras sentenças.

(10) a. João estuda e João trabalha.
b. João abraçou Maria e João beijou Maria.
c. João mostrou o livro pra Maria e João deu o livro pra Maria.
d. João abraçou Maria e João beijou Paula.
e. João é médico e João é psicólogo.
f. João está alegre e João está orgulhoso da Maria.
g. João esteve em Paris e João esteve em Londres.
h. João estuda e Maria estuda.

As versões reduzidas em (9) seriam, então, obtidas através de transformações sintáticas a partir das sentenças em (10), mas essas transformações não teriam impacto semântico, ou seja, o *input* para o componente semântico seriam as versões não reduzidas em (10). E para a interpretação dessas sentenças, nossa entrada lexical de tipo $\langle t, \langle t, t\rangle\rangle$ é suficiente.

Há problemas, no entanto, com esse processo de redução sintática. Nem todos os casos de coordenação são equivalentes às suas versões não reduzidas. Considere, por exemplo, a sentença em (11) e sua versão não reduzida em (12):

(11) Alguém estuda e trabalha.

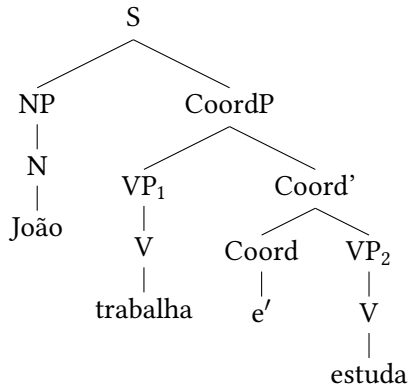
(12) Alguém estuda e alguém trabalha.

Ainda não dissemos nada sobre o pronome indefinido *alguém*, e só iremos fazê-lo mais tarde no capítulo dedicado à análise da quantificação. Mas basta refletirmos um pouco sobre as condições de verdade das duas sentenças acima para notarmos que elas não são equivalentes. Para (11) ser verdadeira, é necessária a existência de uma mesma pessoa que estude e também trabalhe. Já no caso de (12), basta haver uma pessoa que estude e uma que trabalhe, sem a necessidade

de que elas sejam a mesma pessoa. Não queremos, portanto, que o *input* para a interpretação de (11) seja (12), já que elas não têm o mesmo significado.

Busquemos, então, uma alternativa para a análise baseada em redução sintática. Vamos manter que a estrutura das sentenças em (9) que serve de *input* ao componente semântico envolve mesmo a coordenação de constituintes menores que a sentença.

A maneira mais óbvia de concretizar essa abordagem é assumir que existem várias conjunções *e*, cada uma com um tipo semântico diferente. Assim, além da versão de tipo $\langle t, \langle t, t\rangle\rangle$, utilizada na coordenação de sentenças, haveria, por exemplo, uma outra de tipo $\langle et, \langle et, et\rangle\rangle$ utilizada na coordenação de constituintes com denotações de tipo $\langle e, t\rangle$, como em (9a).



Vamos chamar essa outra conjunção de e' para diferenciá-la do e que coordena sentenças. Queremos que essa nova conjunção tome as extensões dos dois VPs (duas funções de tipo $\langle e, t\rangle$) como argumentos e retorne uma nova função desse mesmo tipo $\langle e, t\rangle$. Essa nova função tomará um indivíduo como argumento (fornecido pela extensão do NP sujeito) e aplicará as extensões dos VPs originais a ela. A entrada lexical abaixo formaliza essa intuição:

$$[\![\text{e}']\!] = \lambda F_{\langle e,t\rangle}.\lambda G_{\langle e,t\rangle}.\lambda x_e.\ F'(x)\ \&\ G'(x)$$

Nessa nova entrada, F e G correspondem às extensões dos VPs coordenados e x à extensão do NP sujeito. Com isso em mente, acompanhe atentamente a derivação das condições de verdade da estrutura acima e note que a nova entrada lexical fornece o resultado desejado.

$[\![\text{Coord}']\!] = [\![\text{Coord}]\!]([\![\text{VP}_2]\!])$

$= (\lambda F.\lambda G.\lambda x.\ F'(x)\ \&\ G'(x))([\![\text{VP}_2]\!])$

Como já sabemos que:

$[\![\text{VP}_2]\!] = \lambda x.\ \textsc{estuda}(x)$

e que, portanto:

$[\![\text{VP}_2]\!](x) = 1$ *sse* $\textsc{estuda}(x)$

temos que:

$[\![\text{Coord}']\!] = \lambda G.\lambda x.\ \textsc{estuda}(x)\ \&\ G'(x)$

Quanto a CoordP, valemo-nos de aplicação funcional novamente:

$[\![\text{CoordP}]\!] = [\![\text{Coord}']\!]([\![\text{VP}_1]\!])$

$= (\lambda G.\lambda x.\ \textsc{estuda}(x)\ \&\ G'(x))([\![\text{VP}_1]\!])$

Como já sabemos que:

$[\![\text{VP}_1]\!] = \lambda x.\ \textsc{trabalha}(x)$

e que, portanto:

$[\![\text{VP}_1]\!](x) = 1$ *sse* $\textsc{trabalha}(x)$

temos que:

$[\![\text{CoordP}]\!] = \lambda x_e.\ \textsc{estuda}(x)\ \&\ \textsc{trabalha}(x)$

Note que essa é uma função de tipo $\langle e, t\rangle$ que leva um indivíduo x ao valor de verdade 1 se, e somente se, x estuda e x trabalha. Note, ainda, que dada a já salientada simetria do conectivo lógico &, podemos igualmente escrever:

$[\![\text{CoordP}]\!] = \lambda x_e.\ \textsc{trabalha}(x)\ \&\ \textsc{estuda}(x)$

Finalmente, temos que:

$[\![\text{S}]\!] = [\![\text{CoordP}]\!]([\![\text{NP}]\!])$

$[\![\text{S}]\!] = (\lambda x.\ \textsc{trabalha}(x)\ \&\ \textsc{estuda}(x))(\textit{joão})$

$[\![\text{S}]\!] = 1$ *sse* $\textsc{trabalha}(\textit{joão})\ \&\ \textsc{estuda}(\textit{joão})$

A entrada lexical proposta acima para e' pode ser utilizada não apenas em (9a), mas também em (9d)-(9g), já que todos estes casos envolvem coordenação de constituintes cujas denotações são funções de tipo $\langle e, t\rangle$.

Para casos envolvendo a coordenação de verbos transitivos e de verbos bitransitivos, precisamos de outras entradas lexicais, já que suas denotações são funções de tipo $\langle e, \langle e, t\rangle\rangle$ e $\langle e, \langle e, \langle e, t\rangle\rangle\rangle$ respectivamente. Mas a lógica por trás da definição dessas novas entradas lexicais é semelhante à que utilizamos para e'. Queremos uma função que, após ser aplicada às extensões dos verbos coordenados, retorne como valor uma função do mesmo tipo que as extensões desses verbos. Já os argumentos fornecidos a essa nova função serão, por assim dizer, transmitidos às extensões dos verbos coordenados, como exemplificado abaixo:

$[\![\text{e}''\!]\!] = \lambda F_{\langle e,\langle e,t\rangle\rangle}.\lambda G_{\langle e,\langle e,t\rangle\rangle}.\lambda x_e.\lambda y_e.\ F'(x)(y)\ \&\ G'(x)(y)$

$[\![\text{e}'''\!]\!] = \lambda F_{\langle e,\langle e,\langle e,t\rangle\rangle\rangle}.\lambda G_{\langle e,\langle e,\langle e,t\rangle\rangle\rangle}.\lambda x_e.\lambda y_e.\lambda z_e.\ F'(x)(y)(z) = 1\ \&\ G'(x)(y)(z) = 1$

Vistas sob um certo ângulo, essa entradas lexicais caracterizam uma espécie de economia de meios, em que um mesmo recurso sintático (sujeito, objeto direto, ...) é utilizado duas vezes semanticamente.

O procedimento de postular múltiplas entradas lexicais para a conjunção *e* equivale a assumir que essa conjunção é ambígua, o que, por sua vez, equivale a dizer que existem várias conjunções com significados diferentes, porém homófonas. Ainda que tal procedimento conduza a resultados descritivamente corretos, causa um certo desconforto o fato de que intuitivamente estamos diante de uma mesma conjunção que introduz sempre a ideia de adição, seja essa a adição de proposições, propriedades ou relações. Além disso, essa suposta ambiguidade não é exclusiva do português, existindo em uma variedade de línguas. Tudo isso leva a crer que essa flexibilidade no uso da conjunção *e* não é acidental como

seria de se esperar de uma abordagem como a descrita acima, baseada na simples listagem de múltiplas entradas lexicais de itens homófonos. Seria interessante se pudéssemos relacionar explicitamente todas essas entradas lexicais, de modo que a partir de uma definição básica, derivássemos todas as demais.

Há uma maneira relativamente simples de concretizar essa proposta. Ela se baseia em uma semelhança na forma das entradas lexicais que propusemos acima e que repetimos abaixo por conveniência:

$[\![\text{e}]\!] = \lambda p_t.\ \lambda q_t.\ p = 1 \ \&\ q = 1$

$[\![\text{e}']\!] = \lambda F_{\langle e,t\rangle}.\lambda G_{\langle e,t\rangle}.\lambda x_e.\ F'(x)\ \&\ G'(x)$

$[\![\text{e}'']\!] = \lambda F_{\langle e,\langle e,t\rangle\rangle}.\lambda G_{\langle e,\langle e,t\rangle\rangle}.\lambda x_e.\lambda y_e.\ F'(x)(y)\ \&\ G'(x)(y)$

$[\![\text{e}''']\!] = \lambda F_{\langle e,\langle e,\langle e,t\rangle\rangle\rangle}.\lambda G_{\langle e,\langle e,\langle e,t\rangle\rangle\rangle}.\lambda x_e.\lambda y_e.\lambda z_e.\ F'(x)(y)(z)\ \&\ G'(x)(y)(z)$

Para $[\![\text{e}]\!]$, valemo-nos simplesmente do conectivo lógico sentencial &, já que estávamos coordenando constituintes sentenciais. Nos demais casos, em que coordenamos constituintes com denotações funcionais, o que fizemos foi alimentar cada uma das funções com os mesmos argumentos e conectar os resultados (na metalinguagem) com o & sentencial. A diferença entre $[\![\text{e}']\!]$, $[\![\text{e}'']\!]$ e $[\![\text{e}''']\!]$ está apenas no número de argumentos necessários para que as funções envolvidas retornem um valor de verdade: 1, 2 e 3, respectivamente. É importante notar que, para procedermos dessa forma, é necessário que, com a aplicação dessas funções a esses argumentos, cheguemos a um valor de verdade (tipo t), caso contrário não poderíamos utilizar o conectivo & na metalinguagem. Portanto, apenas extensões de tipos semânticos que terminam em t podem ser coordenadas seguindo essa receita. Exemplos de constituintes que terminam em t são t, $\langle e, t\rangle$, $\langle e, et\rangle$, $\langle et, et\rangle$ .... Esses tipos terminados em t costumam ser chamados de TIPOS BOOLEANOS, em homenagem a George Boole, um dos pais da lógica e da álgebra modernas. Não são booleanos tipos tais como e, $\langle et, e\rangle$, $\langle e, e\rangle$ ...

Tendo isso em mente, podemos tomar $[\![\text{e}]\!]$ como forma básica, e derivar as demais a partir dela.

$[\![\text{e}]\!] = \lambda p_t.\ \lambda q_t.\ p = 1 \ \&\ q = 1$

$[\![\text{e}']\!] = \lambda F_{\langle e,t\rangle}.\lambda G_{\langle e,t\rangle}.\lambda x_e.\ [\![\text{e}]\!](F(x))(G(x))$

$[\![\text{e}'']\!] = \lambda F_{\langle e,\langle e,t\rangle\rangle}.\lambda G_{\langle e,\langle e,t\rangle\rangle}.\lambda x_e.\lambda y_e.\ [\![\text{e}]\!](F(x)(y))(G(x)(y))$

$[\![\text{e}''']\!] = \lambda F_{\langle e,\langle e,\langle e,t\rangle\rangle\rangle}.\lambda G_{\langle e,\langle e,\langle e,t\rangle\rangle\rangle}.\lambda x_e.\lambda y_e.\lambda z_e.\ [\![\text{e}]\!](G(x)(y)(z))(G(x)(y)(z))$

Dessa forma, continuaremos assumindo que a conjunção *e* está associada a múltiplos tipos semânticos. Entretanto, relacionaremos explicitamente as múltiplas extensões envolvidas. Esse procedimento costuma ser chamado de polissemia lógica ou POLIMORFISMO.

Generalizando o que acabamos de ver para todos os tipos booleanos, podemos definir o seguinte esquema para a conjunção:

Versão Polimórfica da Conjunção *e*:

Caso básico: se $a = t$,

$$[\![e_{\langle a,\langle a,a\rangle\rangle}]\!] = \lambda p_t.\ \lambda q_t.\ p = 1 \ \&\ q = 1$$

Casos Derivados: se $a = \langle a_1, \langle a_2, ..., \langle a_n, t\rangle ...\rangle\rangle$ $(n \geq 1)$,

$$[\![e_{\langle a,\langle a,a\rangle\rangle}]\!] = \lambda f_a.\lambda g_a.\lambda\gamma^1_{a_1}.\lambda\gamma^2_{a_2}...\lambda\gamma^n_{a_n}.\ [\![e_{\langle t\langle t,t\rangle\rangle}]\!](f_a(\gamma^1_{a_1})(\gamma^2_{a_2})...(\gamma^n_{a_n}))(g_a(\gamma^1_{a_1})(\gamma^2_{a_2})...(\gamma^n_{a_n}))$$

Dado um caso concreto, basta selecionar a versão correspondente do tipo a no esquema acima. Por exemplo, no caso da coordenação de verbos bitransitivos, a será igual a $\langle e, \langle e, \langle e, t\rangle\rangle\rangle$ e $a_1 = a_2 = a_3 = e$. Como o leitor poderá verificar, ao aplicar esses valores no esquema acima, obteremos exatamente a extensão de e''' que vimos anteriormente.

O esquema acima consegue, portanto, dar conta de todos os casos de coordenação listados em (9), com exceção de (9h). Como atribuímos aos nomes próprios o tipo básico *e*, que não é um tipo booleano, tal esquema não pode ser aplicado e nosso sistema não consegue atribuir uma extensão ao constituinte [$_{\text{NP}}$ João e Maria]. Deixaremos, por hora, essa questão em aberto. Voltaremos a ela no capítulo sobre quantificação, quando estaremos em uma melhor posição para resolvê-la.

Uma formalização diferente

Há uma outra maneira de formalizar essa flexibilidade da conjunção *e*, com a

qual vale a pena estar familiarizado. Trata-se de definir recursivamente as múltiplas entradas da conjunção. Para tanto, vamos retomar rapidamente as versões para a coordenação de verbos intransitivos e transitivos que já vimos. Lembremos que, para dar conta da coordenação de verbos intransitivos, como em *Maria fuma e bebe*, valemo-nos da seguinte entrada lexical:

$$[\![\mathrm{e}_{vi}]\!] = \lambda F_{\langle e,t\rangle}.\lambda G_{\langle e,t\rangle}.\lambda x_e.\ F'(x)\ \&\ G'(x)$$

E para dar conta da coordenação de verbos transitivos, como em *Maria beijou e abraçou Pedro*, valemo-nos de outra extensão:

$$[\![\mathrm{e}_{vt}]\!] = \lambda F_{\langle e,\langle e,t\rangle\rangle}.\lambda G_{\langle e,\langle e,t\rangle\rangle}.\lambda x_e.\lambda y_e.\ F'(x)(y)\ \&\ G'(x)(y)$$

Atente agora para o seguinte: os três primeiros argumentos que a entrada lexical $[\![\mathrm{e}_{vt}]\!]$ toma correspondem às extensões de dois verbos transitivos, *F* e *G*, ambas de tipo $\langle e, et\rangle$ e à extensão do objeto direto, *x*, de tipo *e*. Mas com estas extensões à nossa disposição, podemos construir, por assim dizer, duas extensões de tipo $\langle e, t\rangle$, bastando aplicar *F* e *G* a *x*. Os resultados, *F(x)* e *G(x)* são funções de tipo $\langle e, t\rangle$. Podemos então aplicar a estas novas funções a extensão $[\![\mathrm{e}_{vi}]\!]$ e obteremos o mesmo resultado que teríamos obtido aplicando $[\![\mathrm{e}_{vi}]\!]$ a *F*, *G* e *x* nesta ordem. Em termos formais:

$$[\![e_{\mathrm{VT}}]\!](F)(G)(x) = [\![e_{\mathrm{VI}}]\!](F(x))(G(x))$$

Que esta igualdade é verdadeira pode ser facilmente verificado a partir de $[\![\mathrm{e}_{vt}]\!]$ e $[\![\mathrm{e}_{vi}]\!]$. O que fizemos acima foi redefinir $[\![\mathrm{e}_{vt}]\!]$ a partir de $[\![\mathrm{e}_{vi}]\!]$:

$$[\![e_{\mathrm{VT}}]\!] = \lambda F_{\langle et\rangle}.\lambda G_{\langle et\rangle}.\lambda x_e.\ [\![e_{\mathrm{VI}}]\!](F(x))(G(x))$$

O procedimento que acabamos de ilustrar pode ser generalizado, tornando possível definir a extensão de uma conjunção que toma argumentos de tipo $\langle \alpha, \beta\rangle$ a partir da extensão da conjunção que toma argumentos de tipo β. E, se β for também um tipo funcional $\langle \gamma, \delta\rangle$, definir a conjunção que toma argumentos de tipo $\langle \gamma, \delta\rangle$ a partir da extensão da conjunção que toma argumentos de tipo γ, e assim por diante. Se estivermos lidando apenas com tipos booleanos, essa cadeia irá fatalmente terminar na conjunção que coordena sentenças (tipo t), a qual podemos tomar como sendo básica.

Formalizando tudo isso, chegamos à DEFINIÇÃO RECURSIVA abaixo, em que a é

um tipo booleano:

Versão Recursiva da Conjunção *e*

$$[\![e_{\langle a,\langle a,a\rangle\rangle}]\!] = \begin{cases} \lambda p_t.\ \lambda q_t.\ p = 1 \ \&\ q = 1 & \text{se } a = t \\ \lambda F_{\langle b,c\rangle}.\lambda G_{\langle b,c\rangle}.\lambda x_b.\ [\![e_{\langle c\langle c,c\rangle\rangle}]\!](F(x))(G(x)) & \text{se } a = \langle b, c\rangle \end{cases}$$

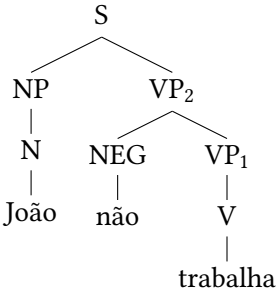
Definições recursivas são caracterizadas por uma ou mais definições básicas (o caso $a = t$ acima) e por uma receita para reduzir casos complexos (os tipos funcionais, na definição acima) em casos cada vez mais simples até que eventualmente se chegue a um caso básico.

3.3 Negação

Considere a sentença (13) abaixo, à qual atribuiremos a estrutura em (14):

(13) João não trabalha.

(14)



Intuitivamente, uma sentença negativa formada com a palavra *não* é verdadeira se a sentença afirmativa correspondente (*João trabalha*, no caso acima) for falsa; e falsa, se a sentença afirmativa for verdadeira. No exemplo acima, sabemos que a extensão de VP_1 é uma função que leva um indivíduo x no valor de verdade 1 se x trabalha, ou no valor de verdade 0, se x não trabalha. O que a negação parece fazer é transformar essa função em uma outra função que, ao receber x como input retorna o valor 1 , se x não trabalha, ou 0, se x trabalha. Para captar essa ideia, vamos assumir a seguinte entrada lexical para a palavra *não*:

$$[\![\text{não}]\!] = \lambda F_{\langle e,t\rangle}.\ \lambda x_e.\ F(x) = 0$$

Ou, equivalentemente, na notação com a negação de predicados:

$[\![\text{não}]\!] = \lambda F_{\langle e,t\rangle}.\ \lambda x_e.\ \neg F'(x)$

Essa extensão é uma função de tipo $\langle\langle$e,t$\rangle$,$\langle$e,t$\rangle\rangle$. Note que o argumento dessa função é, ele mesmo, uma outra função. Vamos calcular o significado de (13)-(14) e verificar se a entrada lexical que acabamos de propor fornece resultados adequados ou não. Desta vez, procederemos de baixo para cima na estrutura (14). Já sabemos a extensão de VP_1:

$[\![\text{VP}_1]\!] = [\![\text{V}]\!] = [\![\text{trabalha}]\!] = \lambda x_e.\ \textsc{trabalha}(x)$

Para obter a extensão de VP_2, podemos usar aplicação funcional, já que a extensão de NEG é uma função cujo domínio é $\text{D}_{\langle e,t\rangle}$, justamente o domínio a que pertence a extensão de VP_1.

$[\![\text{NEG}]\!] = [\![\text{não}]\!] = \lambda F_{\langle e,t\rangle}.\ \lambda x_e.\ \neg F'(x)$

$[\![\text{VP}_2]\!] = [\![\text{NEG}]\!]([\![\text{VP}_1]\!])$
$[\![\text{VP}_2]\!] = (\lambda F_{\langle e,t\rangle}.\ \lambda x_e.\ \neg F'(x))([\![\text{VP}_1]\!])$

Como sabemos que a extensão de VP_1 é uma função que retorna o valor 1 se João trabalha e zero se João não trabalha, temos que:

$[\![\text{VP}_2]\!] = \lambda x_e.\ \neg\textsc{trabalha}(x)$

Para chegarmos às condições de verdade de S, basta utilizarmos aplicação funcional uma vez mais, já que a extensão de NP é de tipo *e*, podendo servir como argumento para a extensão de VP_2.

$[\![\text{NP}]\!] = [\![\text{N}]\!] = [\![\text{João}]\!] = \textit{joão}$

$[\![\text{S}]\!] = [\![\text{VP}_2]\!]([\![\text{NP}]\!])$
$= (\lambda x_e.\ \neg\textsc{trabalha}(x))(\textit{joão})$
$= 1$ *sse* $\neg\textsc{trabalha}(\textit{joão})$

Derivamos, assim, o resultado desejado, confirmando a adequação da entrada lexical que propusemos acima para a negação.

A exemplo do que fizemos no caso da conjunção *e* na seção anterior, podemos nos perguntar se há casos em que a negação se aplica a outros tipos de constituintes que não VPs. Exemplos como os em (15) abaixo não são convincentes, já que, apesar de a negação aparecer junto a verbos transitivos e bitransitivos, eles podem ser analisados como em (16) com a negação formando um constituinte com os respectivos VPs:

(15) a. João não beijou Maria
b. João não apresentou Pedro pra Maria.

(16) a. [João [não [$_{\text{VP}}$ beijou Maria]]]
b. [João [não [$_{\text{VP}}$ apresentou Pedro pra Maria]]]

Para esses casos, portanto, basta a entrada lexical que já propusemos para a negação. Porém, há casos mais complexos envolvendo coordenação e negação em uma mesma sentença que nos fazem crer na possibilidade de a negação estar mesmo formando um constituinte com verbos e não com os respectivos sintagmas verbais. Considere, por exemplo, as sentenças abaixo:

(17) a. João não beijou e não abraçou Maria.
b. João não mostrou e não apresentou o Pedro pra Maria.

Dada a presença de duas negações em cada uma das sentenças acima e dada a maneira como analisamos na seção anterior suas contrapartes sem as negações, parece que estamos, de fato, diante de casos envolvendo negação formando um constituinte com o verbo.

(18) a. [João [[não beijou] e [não abraçou]] Maria]
b. [João [[não mostrou] e [não apresentou]] Pedro pra Maria]

Se essa for mesmo a análise sintática correta para essas sentenças, nosso sistema precisa ser reformulado, já que a extensão que foi atribuída à negação (de tipo $\langle\langle e,t\rangle,\langle e,t\rangle\rangle$) não permite que essa combine diretamente com as extensões de verbos transitivos e bitransitivos (tipos $\langle e,\langle e,t\rangle\rangle$ e $\langle e,\langle e,\langle e,t\rangle\rangle\rangle$, respectivamente).

Há ainda casos em que a negação parece se aplicar a constituintes sentenciais (de tipo t, portanto) e que também não podem ser analisados com a entrada lexical atual da negação. Como possíveis exemplos, temos sentenças com verbos meteorológicos e alguns casos de inversão sujeito-verbo:

(19) a. Não está chovendo.
b. [não [está chovendo]]

(20) a. Não chegaram cartas pra mim ontem.
b. [não [chegaram cartas pra mim ontem]]

Para esses casos, precisaríamos de uma entrada de tipo $\langle t, t \rangle$ para a negação. Tal entrada deve inverter o valor de verdade do constituinte que a negação toma como argumento:

$[\![\text{não}_{\langle t,t \rangle}]\!] = (\lambda p_t .\ p = 0)$

O papel semântico dessa negação sentencial pode também ser representado em uma tabela como a abaixo:

(21) Tabela de verdade da negação

ϕ	não ϕ
1	0
0	1

De acordo com a entrada acima, a negação toma um valor de verdade p como argumento e retorna 1 se p for igual a 0, e 0 se p for igual a 1. Aplicando-a ao exemplo em (19), temos:

[[não está chovendo]] = [[não]]([[está chovendo]])
[[não está chovendo]] = 1 *sse* [[está chovendo]] = 0
[[não está chovendo]] = 1 *sse* não estiver chovendo

Essa flexibilidade da negação em se associar a constituintes de diversos tipos semânticos é semelhante ao que vimos com a conjunção *e*. Assim, munidos da estratégia que utilizamos nos casos de coordenação, podemos pensar em associar uma multiplicidade de tipos à negação, definidos a partir de um tipo básico, de modo a relacionar seus diversos usos. O esquema seguinte, construído à maneira do esquema proposto para a conjunção *e* para tipos booleanos a, faz justamente isso:

Versão Polimórfica da Negação *não*:

Caso básico: Se $a = t$,

$$[\![\text{não}_{\langle a,a\rangle}]\!] = \lambda p_t.\ p = 0$$

Casos Derivados: Se $a = \langle a_1, \langle a_2, ..., \langle a_n, t\rangle\rangle\rangle$ $(n \geq 1)$,

$$[\![\text{não}_{\langle a,a\rangle}]\!] = \lambda F_a.\lambda\gamma^1_{a_1}.\lambda\gamma^2_{a_2}...\lambda\gamma^n_{a_n}.\ [\![\text{não}_{\langle t,t\rangle}]\!](F_a(\gamma^1_{a_1})(\gamma^2_{a_2})...(\gamma^n_{a_n}))$$

Note que o tipo básico é $\langle t, t\rangle$. Para obtermos nossa entrada lexical original de tipo $\langle et, et\rangle$, basta tomarmos *a* como sendo o tipo $\langle e, t\rangle$ e a_1 o tipo e:

$$[\![\text{não}_{\langle et,et\rangle}]\!] = \lambda F_{\langle e,t\rangle}.\lambda x_e.\ [\![\text{não}_{\langle t,t\rangle}]\!](F_{\langle e,t\rangle}(x_e))$$

$$[\![\text{não}_{\langle et,et\rangle}]\!] = \lambda F_{\langle e,t\rangle}.\lambda x_e.\ (\lambda p_t.\ p = 0)(F(x))$$

$$[\![\text{não}_{\langle et,et\rangle}]\!] = \lambda F_{\langle e,t\rangle}.\lambda x_e.\ F(x) = 0$$

$$[\![\text{não}_{\langle et,et\rangle}]\!] = \lambda F_{\langle e,t\rangle}.\lambda x_e.\ \neg F'(x)$$

De maneira análoga, obtemos os demais tipos, bastando aplicar o esquema acima a eles.

Também a exemplo do que vimos para a conjunção *e*, podemos formalizar as múltiplas extensões da negação através de uma definição recursiva. Nesse caso, a ideia é que, quando o argumento da negação for de tipo funcional $\langle b, c\rangle$, podemos definir a extensão correspondente $[\![\text{não}_{\langle bc,bc\rangle}]\!]$ em função da extensão que toma argumentos de tipo c $[\![\text{não}_{\langle c,c\rangle}]\!]$. Se c for também um tipo funcional, repetimos o procedimento até chegarmos à versão básica de tipo $\langle t, t\rangle$. Lembre-se que estamos lidando com tipos booleanos e que, portanto, esse processo sempre terminará no caso básico. Em termos formais:

Versão Recursiva da negação

$$[\![\text{não}_{\langle a,a\rangle}]\!] = \begin{cases} \lambda p_t.\ \ p = 0 & \text{se } a = t \\ \lambda F_{\langle b,c\rangle}.\lambda x_b.\ [\![\text{não}_{\langle c,c\rangle}]\!](F(x)) & \text{se } a = \langle b, c\rangle \end{cases}$$

Fica a cargo do leitor como exercício técnico aplicar a definição acima para obter as negações correspondentes a verbos intransitivos, transitivos e bitransitivos.

Sugestões de leitura

Para um tratamento lógico da coordenação e da negação vistas como operadores sentenciais, ver Gamut (1991), volume 1. Para uma boa introdução à lógica em português, ver Mortari (2016). Para discrepâncias entre a interpretação puramente verifuncional que a lógica clássica confere a operadores sentenciais como *e*, *ou*, *se*, *não* e a interpretação e uso de suas contrapartes nas línguas naturais, ver Grice (1989), Gamut (1991). Para um tratamento flexível da coordenação, consultar Gazdar (1980) e Partee & Rooth (1983). Para um estudo erudito e abrangente sobre a negação, ver Horn (1989).

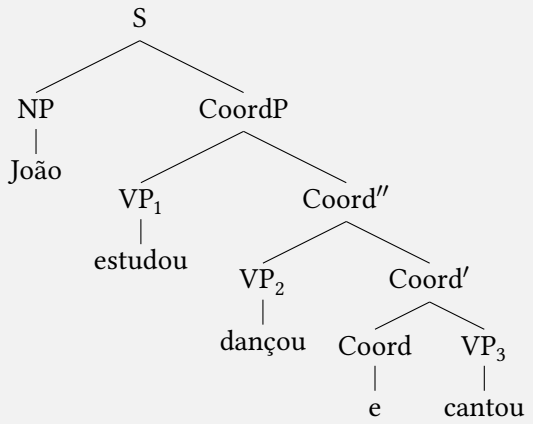
Exercícios

I. Aplique a definição recursiva proposta para a conjunção *e* a um caso envolvendo coordenação de verbos bitransitivos, como em *João mostrou e deu o livro para Maria*. Use a extensão resultante e calcule as condições de verdade dessa sentença.

II. Considere a sentença abaixo, que envolve a coordenação de três VPs:

(1) João estudou, dançou e cantou.

Assuma a seguinte estrutura sintática para essa sentença:

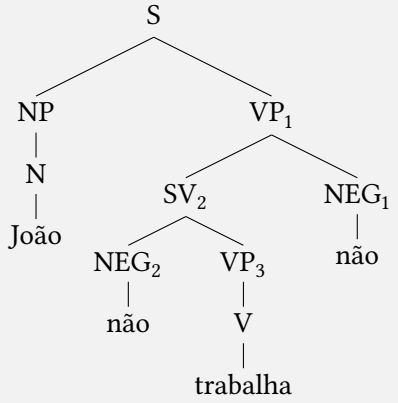


Proponha uma entrada lexical adicional para a conjunção *e* que permita ao nosso sistema interpretar a estrutura acima e mostre, passo a passo, como obter suas condições de verdade.

III. Considere a sentença abaixo, que contém duas negações, um tipo de sentença bastante comum no português falado no Brasil.

(1) João não trabalha não.

Assuma para essa sentença a estrutura sintática abaixo:



Sua tarefa é derivar as condições de verdade que o nosso sistema prevê para esta estrutura. Você deve proceder passo a passo indicando sempre

quais os princípios composicionais utilizados. Feito isso, reflita sobre as condições de verdade obtidas e diga se elas fazem jus ou não ao significado dessa sentença. Se você achar que as condições de verdade obtidas não estão corretas, arrisque um palpite sobre o que pode estar acontecendo.

IV. A sentença abaixo apresenta um caso de ambiguidade estrutural.

(1) João não trabalha e estuda.

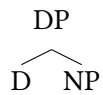
Desenhe as duas estruturas possíveis e as respectivas derivações semânticas, de acordo com o que vimos neste capítulo sobre coordenação e negação.

4 Referência

Neste capítulo, analisaremos expressões cujas extensões são indivíduos. Chamaremos estas expressões de EXPRESSÕES REFERENCIAIS. Voltaremos a falar sobre nomes próprios, e discutiremos descrições definidas e pronomes. Nosso léxico será ampliado com a introdução do artigo definido singular (*o*, *a*) e de pronomes pessoais como *ele* e *ela*.

4.1 Descrições definidas

Descrições definidas são expressões introduzidas pelo artigo definido (*o*, *a*, *os*, *as*) seguido de um sintagma nominal, como em *o livro*, *a capital de Alagoas*, *os jogadores do Palmeiras* ou *as praias de Santa Catarina*. Do ponto de vista sintático, vamos categorizar o artigo definido como sendo um determinante (D), que toma como complemento um sintagma nominal (NP), sendo a projeção resultante um sintagma determinante (DP):



NPs, a partir de agora, são estruturas nucleadas por um substantivo (nome), sem a presença do artigo. Nos exemplos do parágrafo anterior, são NPs as sequências *livro*, *capital de Alagoas*, *jogadores do Palmeiras* e *praias de Santa Catarina*. Voltaremos aos nomes próprios mais adiante.

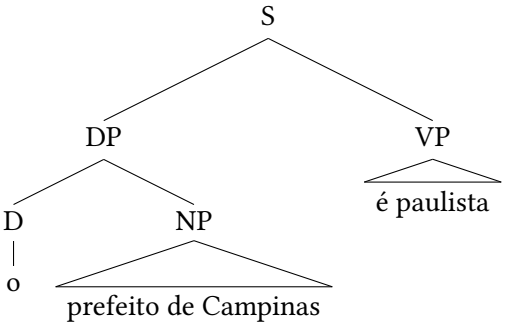
Passemos, então, ao significado de uma descrição definida. Limitaremos nossa atenção às descrições definidas singulares, comparando-as com o que já vimos sobre os nomes próprios (também singulares). Assim como no caso dos nomes próprios, valemo-nos a todo instante de descrições definidas para falarmos de objetos, pessoas, lugares e eventos. Por exemplo, empregamos a descrição definida *o prefeito de Campinas* na sentença abaixo para falarmos da pessoa que ocupa atualmente este cargo político.

(1) O prefeito de Campinas é paulista.

Se o atual prefeito de Campinas é João da Silva (JS), então (1) é verdadeira se JS é paulista e falsa se JS não é paulista. Iremos captar este fato, assumindo que a extensão de um DP definido é o indivíduo que satisfaz a descrição apresentada pelo NP que segue o artigo definido.

$[\![[_{\text{DP}}$ o prefeito de Campinas $]]\!]$ = o indivíduo *a*, tal que *a* é prefeito de Campinas

Qual a extensão do artigo definido *o*? Olhemos para a estrutura (simplificada) de (1) dada abaixo:



A extensão do NP *prefeito de Campinas* é uma função que leva um indivíduo *x* no valor de verdade 1, se *x* é prefeito de Campinas, e no valor de verdade 0, se *x* não é prefeito de Campinas. No cenário que estamos considerando, esta função leva JS no valor 1 e todos os demais indivíduos no valor 0. O que o artigo definido parece fazer, então, é olhar para a extensão de NP e selecionar o indivíduo ao qual esta função atribui valor 1. Isso é o que está formalizado na entrada lexical abaixo (que ainda sofrerá modificações):

$[\![\text{o}]\!] = \lambda F_{\langle e,t\rangle}$. o indivíduo *a*, tal que $F(a) = 1$

Há uma notação compacta para representar o único indivíduo que satisfaz uma certa condição ϕ, que é $\iota x[\phi]$. Valendo-nos dela, e da notação de predicados, podemos representar a extensão de *o* da seguinte forma:

$[\![\text{o}]\!] = \lambda F_{\langle e,t\rangle}$. $\iota x[F'(x)]$

A extensão do artigo definido é, portanto, uma função de tipo $\langle\langle e,t\rangle, e\rangle$. Como a extensão de NP é de tipo $\langle e,t\rangle$, utilizamos aplicação funcional para obter a ex-

tensão do DP sujeito em (1), que será de tipo *e*.

$[\![[_{\text{DP}}$ o prefeito de Campinas $]]\!]$ = $[\![$o$]\!]([\![[_{\text{NP}}$ prefeito de Campinas $]]\!])$
= $\iota x[[\![\text{NP}]\!](x) = 1]$
= $\iota x[$ PREFEITO(*x, Campinas*)]
= o indivíduo *x*, tal que *x* é prefeito de Campinas

Como a extensão de VP é de tipo $\langle e, t\rangle$, utilizando aplicação funcional mais uma vez, chegamos às condições de verdade de (1).

$[\![(1)]\!]$ = 1 *sse* PAULISTA(ιx[PREFEITO(*x, Campinas*)])
$[\![(1)]\!]$ = 1 *sse* o indivíduo *x*, tal que *x* é prefeito de Campinas, é paulista

4.1.1 Unicidade

As condições de verdade acima parecem adequadas. A sentença (1) será, de fato, verdadeira, se o indivíduo que ocupa atualmente o cargo de prefeito de Campinas, seja ele quem for, for paulista, e falsa se este indivíduo não for paulista. Nossa entrada lexical para o artigo definido singular parece correta.

Consideremos agora as sentenças abaixo, e perguntemo-nos se elas são verdadeiras ou falsas:

(2) O presidente de Campinas é paulista.

(3) O vereador de Campinas é paulista.

Certamente, não queremos dizer que (2) e (3) são verdadeiras. Seriam elas falsas? Mas, se esse fosse o caso, preveríamos que as negações dessas sentenças fosses verdadeiras:

(4) O presidente de Campinas não é paulista.

(5) O vereador de Campinas não é paulista.

Entretanto, compare essas sentenças com a contraparte negativa da sentença (1):

(6) O prefeito de Campinas não é paulista.

Enquanto (6) é a coisa certa a se dizer se acreditamos que (1) seja falsa, o uso de (4) e (5) é no mínimo estranho. Na verdade, o uso destas contrapartes negativas parece tão inadequado quanto o uso de (2) e (3).

O que parece problemático com o uso das descrições definidas em (2)-(5) é que as descrições introduzidas pelos NPs *presidente de Campinas* e *vereador de Campinas* não se aplicam a um único indivíduo. Por um lado, não existe presidente de Campinas, e por outro existem vários vereadores na cidade. E de fato, uma pessoa que use (2) e (4) parece pressupor, tomar como certo, que exista um presidente de Campinas. Já uma pessoa que use (3) ou (5) parece tomar como certo que Campinas possua apenas um vereador.

Em termos mais formais, as extensões dos NPs *presidente de Campinas* e *vereador de Campinas* não retornam o valor de verdade 1 para um único indivíduo. No caso de *presidente de Campinas* todo indivíduo é levado ao valor de verdade 0, e no caso de *vereador de Campinas* mais de um indivíduo é levado ao valor de verdade 1. O exemplo que havíamos discutido mais acima envolvendo o NP *prefeito de Campinas* não era problemático, justamente porque a extensão deste NP retorna o valor 1 para um, e somente um, indivíduo.

Para captar esses fatos, vamos reformular a entrada lexical do artigo definido. Nossa entrada atual diz que o artigo definido é uma função que toma como argumento funções de tipo $\langle e, t\rangle$. Até aqui não impusemos nenhuma condição sobre essas funções. É o que faremos agora. Vamos assumir que o domínio da extensão do artigo definido contém apenas funções de tipo $\langle e, t\rangle$ que retornam o valor 1 para um único indivíduo e 0 para todos os demais. Para representar esse tipo de função, vamos nos valer na metalinguagem da expressão $\exists!x$, que deve ser lida como 'existe um único *x*'. Vejamos:

$$[\![\text{o}]\!] = \lambda F_{\langle e,t\rangle} : \exists!x \in D_e[F'(x)].\ \iota x[F'(x)]$$

A fórmula $\underline{\exists!x \in D_e[F'(x)]}$ que aparece na entrada acima indica uma condição sobre o domínio da extensão do artigo definido. Dizemos que essa extensão é uma FUNÇÃO PARCIAL de tipo $\langle\langle e, t\rangle, e\rangle$, já que seu domínio é um subconjunto de $D_{\langle e,t\rangle}$. Para pertencer ao domínio dessa função, não basta ser uma função de tipo $\langle e, t\rangle$. É preciso ser uma função que retorne o valor 1 para um único indivíduo.

Vejamos as consequências disso para o nosso sistema. Se voltarmos a (2)-(5), e tentarmos derivar a extensão de DP utilizando aplicação funcional, encontraremos um problema, já que a extensão de NP não pertence ao domínio da extensão do artigo definido. Não havendo como continuar a derivação, o sistema não atribui condições de verdade às sentenças. Podemos dizer para esses casos que a existência de um único presidente ou vereador de Campinas é condição necessária para que as respectivas sentenças recebam um valor de verdade. No caso da sentença (1), que contém o DP *prefeito de Campinas*, a condição de que exista um

único indivíduo que é prefeito de Campinas é satisfeita e a sentença será verdadeira ou falsa, a depender se este indivíduo é ou não paulista. Já nos casos de (2)-(5), as respectivas condições não são satisfeitas e as sentenças não são verdadeiras nem falsas. Podemos dizer que seu uso é infeliz, dada a inexistência de um único presidente ou vereador de Campinas.

Condições como a condição de unicidade imposta pelo artigo definido nos casos acima são chamadas de PRESSUPOSIÇÕES. Analisada do ponto de vista semântico, dizemos que uma sentença S pressupõe p se a verdade de p é condição necessária para que S seja verdadeira ou falsa. Do ponto de vista pragmático, ou seja, do uso da linguagem, dizemos que S pressupõe p se S só pode ser usada adequadamente em contextos em que a verdade de p é tomada como certa, ou seja, como conhecimento compartilhado pelo falante e por sua audiência. Em uma visão ainda mais estritamente pragmática, diz-se que um falante pressupõe p quando ele age tomando como certa a verdade de p.

Tendo isso em mente, não é difícil notar que a condição de unicidade que introduzimos na entrada lexical do artigo definido é, na maioria das vezes, excessivamente restritiva. Por exemplo, imagine que estejamos envolvidos em uma conversa sobre a atual conjuntura política da cidade de Campinas, e que no meio desta conversa alguém diga o seguinte:

(7) O prefeito precisa melhorar o transporte municipal.

De acordo com o que dissemos acima, deveríamos esperar que o uso de (7) nesse contexto fosse infeliz, já que a mesma pressupõe a existência de um único prefeito no mundo, e esse, obviamente, não é o caso. Entretanto, o uso de (7) na situação acima é perfeitamente natural. A razão é que o contexto deixa claro que estamos falando do prefeito de Campinas. Vamos dizer que o único indivíduo saliente no contexto que ocupa o cargo de prefeito é o prefeito de Campinas. Se, ao contrário, nossa conversa estivesse girando em torno de uma reunião entre o prefeito de Campinas e o prefeito do Rio de Janeiro, nossa sentença já não seria mais adequada, pois o contexto, neste caso, torna salientes dois indivíduos que são prefeitos. Precisamos então adicionar o conceito de saliência à nossa entrada lexical do artigo definido:

$[\![\text{o}]\!] = \lambda F_{\langle e,t\rangle}$: $\exists!x_e$ *saliente no contexto, tal que* $[F'(x)]$. $\iota x[F'(x)$ & x está saliente no contexto]

Especificar exatamente como um indivíduo se torna saliente em um contexto é um dos tópicos mais controversos nos estudos da interface Semântica-Prag-

mática. Contentaremo-nos aqui em notar que o uso de descrições definidas é, como acabamos de ver, sensível a esse conceito, e que a entrada lexical acima busca representar, ainda que de maneira rudimentar, essa sensibilidade. Salvo indicações em contrário, omitiremos essa importante ressalva no restante desse livro.

4.2 Nomes próprios

Temos agora dois tipos de expressões referenciais em nosso sistema: nomes próprios e descrições definidas. Neste ponto, seria interessante comparar as condições de verdade de pares de sentenças cujos membros diferem apenas pelo uso de um nome próprio no lugar de uma descrição definida, ou vice-versa. Comparemos, por exemplo, as condições de verdade de (1), repetida abaixo como (8), com as condições de verdade de (9):

(8) O prefeito de Campinas é paulista.
⟦(8)⟧ = 1 *sse* o indivíduo *a*, tal que *a* é prefeito de Campinas é paulista

(9) João da Silva é paulista.
⟦(9)⟧ = 1 *sse* JS é paulista.

Observe a diferença entre o que está à direita de *sse* em (8) e (9). Esta diferença reflete uma diferença de significado entre as sentenças. Não queremos atribuir a (8) e (9) o mesmo significado, ou seja, as mesmas condições de verdade, já que podemos facilmente imaginar situações em que uma é verdadeira e a outra falsa. Basta que JS não seja prefeito de Campinas. E, mesmo que JS seja prefeito de Campinas, uma pessoa que não esteja ciente desse fato pode muito bem acreditar que (8) seja verdadeira e (9) falsa, ou vice-versa, sem que atribuamos a esta pessoa crenças contraditórias. O que o nosso sistema diz sobre as situações em que JS é, de fato, prefeito de Campinas, é que (8) e (9) têm a mesma extensão, o valor de verdade 1, não o mesmo significado. Obviamente, na origem dessa distinção está a diferença na maneira como apresentamos a extensão do nome próprio João da Silva e a extensão do DP *o prefeito de Campinas*:

⟦João da Silva⟧ = JS
⟦[o prefeito de Campinas]⟧ = o indivíduo *a*, tal que *a* é prefeito de Campinas

Mesmo que, em certas situações, *João da Silva* e *o prefeito de Campinas* tenham a mesma extensão (o indivíduo JS), eles não têm o mesmo significado, e isto está

expresso na diferença que acabamos de enfatizar. A extensão do nome próprio é identificada diretamente com o indivíduo JS, enquanto que a extensão da descrição definida o é através, como o próprio nome já diz, da descrição de uma das propriedades do indivíduo. Com isso, diferentes situações ou estados de coisas podem levar a diferentes extensões para a descrição definida *o prefeito de Campinas*. Se Pedro de Oliveira, em vez de de João da Silva, for o prefeito de Campinas, então a extensão da descrição definida será o indivíduo Pedro de Oliveira, em vez do indivíduo João da Silva. Já no caso do nome próprio *João da Silva*, diferentes situações ou estados de coisas não alteram sua extensão, que é sempre o mesmo indivíduo João da Silva. Uma expressão cuja extensão é fixa, ou seja, que nunca varia de situação para situação, é chamada de DESIGNADOR RÍGIDO. Assim, em nosso sistema, nomes próprios, mas não descrições definidas, são designadores rígidos.

Evidência para essa distinção entre nomes próprios (designadores rígidos) e descrições definidas (designadores não rígidos) pode ser obtida através do seguinte exercício mental: para cada uma das sentenças abaixo, imaginemos diferentes situações em que a sentença seja verdadeira.

(10) O autor de Dom Casmurro era carioca.

(11) Machado de Assis era carioca.

No caso de (10), nossa imaginação pode nos levar a situações em que, por exemplo, José de Alencar tenha nascido no Rio de Janeiro e seja o autor de Dom Casmurro. Ou podemos pensar que Paulo Coelho tenha escrito Dom Casmurro, e que era um carioca do século dezenove. Pouco importa quem imaginamos como sendo o autor de Dom Casmurro. O que há em comum entre essas situações é que o autor de Dom Casmurro, seja ele quem for, nasceu no Rio de Janeiro. Já no caso de (11), todas as situações que imaginamos dizem respeito ao mesmo indivíduo: Machado de Assis. Seja ele escritor ou não, pouco importa. Mas note que temos a intuição de que é sempre do mesmo indivíduo que estamos falando. A ideia é que essa diferença, nas situações hipotéticas que construímos ao imaginar (10) e (11) como verdadeiras, é reflexo do fato de a descrição definida *o autor de Dom Casmurro* não ser um designador rígido, enquanto o nome próprio *Machado de Assis* o é.

Apesar da evidência acima a favor do tratamento dos nomes próprios como designadores rígidos, a análise não escapa de um problema que foi objeto da atenção de vários filósofos e linguistas. Para entendermos a fonte do problema, é preciso nos darmos conta da possibilidade de uma pessoa possuir mais de um nome. Por exemplo, a famosa atriz brasileira conhecida pelo nome de Fernanda

Montenegro, chama-se na verdade Arlette Torres (seu nome de casada). Muitas pessoas, entretanto, não sabem disso. Imagine, agora, uma pessoa que tenha conhecido Fernanda/Arlette antes de ela se tornar atriz e adotar o nome artístico pelo qual é conhecida. Imagine que esta pessoa tenha visto Fernanda/Arlette pela última vez quando ela ainda era uma criança, e que, mesmo tendo visto a atriz atuar nas novelas, jamais tenha se dado conta de que Arlette Torres e Fernanda Montenegro são a mesma pessoa. Por fim, imagine que, num belo dia, alguém chegue para essa pessoa e diga a sentença abaixo:

(12) Arlette Torres é Fernanda Montenegro.

A pessoa em questão certamente se surpreenderia com essa revelação. Até aquele momento, tal pessoa provavelmente acreditava na negação de (12), ou seja em (13):

(13) Arlette Torres não é Fernanda Montenegro.

Mas vejamos que significado, ou seja, que condições de verdade, o nosso sistema atribui a essas sentenças. Não dissemos nada sobre o uso do verbo *ser* em sentenças desse tipo, mas é natural que sua extensão estabeleça a identidade entre as extensões dos nomes presentes na sentença:

$[\![\text{é}]\!] = \lambda x.\ \lambda y.\ x = y$

E quanto aos nomes próprios? Como ambos os nomes são nomes da mesma pessoa, não temos muita escolha:

$[\![\text{Arlette Torres}]\!] = [\![\text{Fernanda Montenegro}]\!] = \textit{arlette torres}$

Lembre-se que o que aparece representado em itálico depois das igualdades é o indivíduo (carne e osso). Assim, poderíamos alternativamente dizer:

$[\![\text{Arlette Torres}]\!] = [\![\text{Fernanda Montenegro}]\!] = \textit{fernanda montenegro}$

Efetuando a derivação das condições de verdade de (12) (a derivação fica a cargo do leitor), obteremos o seguinte:

$[\![(12)]\!] = 1$ *sse* $\textit{arlette torres} = \textit{arlette torres}$

Mas o que aparece após a igualdade é uma tautologia, já que sabemos de antemão que todo indivíduo é idêntico a si mesmo. Se pensarmos melhor, essas são as mesmas condições de verdade que o nosso sistema atribui à sentença abaixo:

(14) Arlete Torres é Arlete Torres.

Essa sentença, de fato, soa como tautológica, e certamente ninguém, incluindo o personagem de nossa história, se surpreenderia ao ouvi-la. Como então explicar a surpresa de nossa personagem diante de (12) e sua indiferença diante de (14), se atribuímos a elas o mesmo significado?

Vendo o problema de outro ângulo: como nossa personagem acredita na verdade de (13), e como nosso sistema atribui a essa sentença as condições de verdade abaixo, deveríamos atribuir a essa pessoa uma crença contraditória, já que tais condições de verdade não são nunca satisfeitas:

$[\![(12)]\!] = 1$ *sse arlette torres* $\neq$ *arlette torres*

Mas acreditar em (13) certamente não implica contradição. Estamos, pois, diante de um impasse gerado pela nossa hipótese de que o valor semântico de um nome próprio é diretamente identificado com o indivíduo portador do nome.

Não iremos investigar aqui possíveis reformulações em nosso sistema para tentar lidar com esse problema. O leitor, entretanto, deve estar ciente disso, bem como entrar em contato com a literatura clássica sobre o tema (ver, para isso, as referências ao final do capítulo).

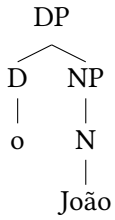
4.2.1 Nomes próprios como predicados

Em todos os nossos exemplos até o momento, os nomes próprios diferem sintaticamente das descrições definidas pelo fato de não virem precedidos de um artigo definido. Isso, entretanto, não é sempre verdade em português, conforme atestam as sentenças abaixo:

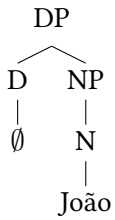
(15) a. O João está sorrindo.
b. O João beijou a Maria.

Casos como estes existem também em dialetos do grego e do alemão, por exemplo. A semelhança sintática entre os nomes próprios em (15) e as descrições definidas que analisamos mais acima parece clara o bastante para que tentemos assimilar essas categorias. Considere, então, a seguinte estrutura para o sintagma

o João:



Considere, ainda, a possibilidade de estender esse tratamento aos casos em que o nome próprio não vem precedido do artigo. Para tanto, assumamos que exista um determinante nulo, idêntico semanticamente ao artigo definido, mas que não é realizado foneticamente, conforme ilustrado abaixo:



Unificaríamos assim a estrutura sintática de constituintes como *o João* e *o menino*, ambos tratados agora como DPs ramificados em D e NP. Criamos, entretanto, um problema para o componente semântico. Nossa entrada lexical para *João* atribui a esse item uma extensão de tipo e. Como a extensão do artigo definido é de tipo $\langle\langle e, t\rangle, e\rangle$, temos uma incompatibilidade de tipos, impossibilitando a obtenção de um extensão para o DP *o João*. Como podemos resolver esse impasse?

Tratamos NPs como *menino* ou *prefeito de Campinas* como predicados com extensões do tipo $\langle e, t\rangle$. A solução mais óbvia para o problema acima é tratarmos nomes próprios também como sendo de tipo $\langle e, t\rangle$. Mas que função um nome como *João* teria como extensão? Uma possibilidade é assumir que a extensão de *João* seja uma função que leva um indivíduo *x* ao valor de verdade 1 se, e somente se, *x* se chama João. A definição exata de como um indivíduo vem a se chamar João não nos interessa aqui. Batismo, registro em cartório, título de nobreza e criação de apelido são apenas algumas das maneiras por meio das quais alguém vem a ser chamado pelo nome que tem. Como falantes de uma língua, sabemos que nomes próprios são usados quando queremos nos referir a uma pessoa que passou por uma dessas, digamos, cerimônias, tendo ali sua pessoa associada a uma expressão verbal. É isso que queremos dizer com a condição ‘x se chama

NN' onde NN é um nome próprio qualquer. De acordo com essa ideia, a entrada lexical de *João* passa a ser a seguinte:

⟦João⟧ = $\lambda x.$ x se chama *João*

Uma vez mais, aplicação funcional nos dá a extensão de *o João* a partir de seus constituintes imediatos, conforme mostrado abaixo:

($\lambda F_{\langle e,t\rangle}$: $\exists!x$ saliente no contexto $[F'(x)]$. $\iota x[F'(x)]$)(⟦[$_{\text{NP}}$João]⟧)
= ιx[x está saliente no contexto & x se chama *João*]

Para obtermos as condições de verdade da sentença (15a), utilizamos aplicação funcional novamente. Desta vez, a extensão de DP, tipo e, serve como argumento para a extensão de VP, tipo $\langle e, t\rangle$.

⟦S⟧ = ⟦[$_{\text{VP}}$está sorrindo]⟧([$_{\text{DP}}$o João])
= 1 *sse* o único indivíduo saliente no contexto que se chama João está sorrindo.

As condições acima parecem adequadas. Note a menção à saliência contextual, semelhante ao caso das descrições definidas que discutimos anteriormente. Podemos apreciar o efeito dessa condição imaginando contextos que tornam salientes mais de um indivíduo com o mesmo nome. Por exemplo, imagine que tenhamos dois amigos comuns, ambos chamados João, e que os dois estejam caminhando em nossa direção, mas apenas um deles esteja sorrindo. Em uma situação como essa, o uso do nome próprio desacompanhado de modificadores, como em *o João está sorrindo*, não é adequado. Nossa análise capta esse fato corretamente. Se sabemos que um dos amigos é carioca e o outro paulista, e que este último é o que está sorrindo, podemos dizer algo como *O João paulista está sorrindo*.

No que diz respeito à condição de unicidade, essa alternativa teórica capta corretamente a semelhança entre nomes próprios e descrições definidas. No entanto, perde-se a distinção importante entre a rigidez dos nomes e a não rigidez das descrições definidas. Não é difícil imaginar diferentes situações em que os indivíduos chamados *João* não são os mesmos. A consequência disso é que a extensão dos DPs *João* ou *O João* passa a variar de situação para situação. Em outras palavras, nomes próprios não são mais designadores rígidos de acordo com essa teoria, o que é problemático, dado o que vimos anteriormente. Vamos aqui deixar em aberto a questão de como reparar essa alternativa teórica de modo a tornar os nomes próprios designadores rígidos.

Consideremos uma outra possibilidade. Tomemos a extensão do nome próprio *João* como sendo a função que leva o indivíduo João no valor de verdade 1 e todos os demais indivíduos no valor de verdade 0:

⟦João⟧ = ($\lambda x.\ x$ = João)

Para o DP *O João*, obteríamos o seguinte:

⟦O João⟧ = o único indivíduo *x*, tal que *x* = *joão*

Como ninguém além do próprio João pode satisfazer a condição acima, não é difícil notar que isso é equivalente ao que tínhamos antes:

⟦O João⟧ = *joão*

Dois pontos importantes: em primeiro lugar, como o único indivíduo idêntico ao João é o próprio João, a condição de unicidade que associamos ao artigo definido será sempre satisfeita. Em segundo lugar, como a função correspondente à extensão do nome próprio é definida mencionando diretamente o único indivíduo que ela leva ao valor de verdade 1, os DPs em questão são designadores rígidos, diferentemente dos DPs que não envolvem nomes próprios. *João* ou *O João* denotam o indivíduo João qualquer que seja a situação em questão. Preservam-se assim tanto as propriedades semânticas associadas aos nomes próprios quanto aquelas associadas às descrições definidas usuais que discutimos mais acima.

Vamos encerrar esse tema por aqui. Como se pode notar, nem mesmo os nomes próprios, que pareciam ser as expressões mais simples semanticamente, estão livres de controvérsias. Pelo contrário, como já salientamos, eles foram e continuam sendo alvos de debates lógicos, linguísticos e filosóficos. Mais uma vez, remetemos o leitor às sugestões de leitura ao final do capítulo.

4.3 Pronomes

Imagine que alguém nos apresente a sentença em (16) e nos pergunte se ela é verdadeira ou falsa:

(16) Ele é italiano.

Nossa reação, claro, será de perplexidade, e provavelmente indagaremos algo como "Mas, ele quem?". Se a pessoa nos disser que está falando do João, ou se ela

simplesmente apontar para o João, assumindo que ele esteja nas proximidades, então diremos que a sentença é verdadeira se o João for italiano, e falsa se ele não for. Tivesse a pessoa apontado para o Pedro e diríamos que a sentença seria verdadeira se o Pedro fosse italiano, e falsa se o Pedro não fosse italiano.

Imagine agora que nos fosse apresentada a sentença em (17):

(17) Ele é tio dele.

Dita, assim, fora de contexto, tal sentença também levaria a indagações, talvez algo como "Espere aí, quem é tio de quem?". Se a pessoa então repetisse a sentença, mas desta vez apontando (enquanto fala) primeiro para o João e logo depois para o Pedro, diríamos que a sentença seria verdadeira se o João fosse tio do Pedro, e falsa se o João não fosse tio do Pedro. Tivesse a pessoa apontado primeiro para o Pedro e depois para o João, diríamos que a sentença seria verdadeira se o Pedro fosse tio do João e falsa se o Pedro não fosse tio do João.

Toda essa discussão está, obviamente, centrada no uso do pronome pessoal *ele*. O que acabamos de ver já basta para tirarmos algumas conclusões a esse respeito. Vamos listá-las:

- A exemplo dos nomes próprios e das descrições definidas, pronomes são usados a todo instante para falar de indivíduos (pessoas, animais, objetos, ...).
- Fora de contexto, não faz sentido perguntar se uma sentença que contém um pronome é verdadeira ou falsa. É só quando estamos cientes do contexto em que a sentença foi dita que podemos nos pronunciar a respeito.
- Da mesma forma, não faz sentido perguntar a quem ou a que um pronome se refere, se não dermos informações sobre o contexto. Diferentes contextos levam a diferentes respostas.
- Sentenças podem conter mais de um pronome e, nesse caso, o contexto de fala deve ser elaborado de modo a deixar claro a quem ou a que estamos nos referindo ao usar cada um dos pronomes. Mesmo dentro de uma mesma sentença, diferentes pronomes podem estar relacionados a diferentes indivíduos.

Refletindo sobre essa lista, podemos extrair algumas direções teóricas sobre a semântica dos pronomes:

- Parece natural assumir que, a exemplo dos nomes próprios e das descrições definidas, a extensão de um pronome é um indivíduo. Pronomes seriam, portanto, expressões referenciais associadas ao tipo semântico *e*.
- No entanto, é preciso relativizar a extensão de um pronome a um parâmetro que traduza de alguma forma o papel do contexto de fala.
- Da mesma forma, é preciso relativizar a extensão das sentenças que contêm pronomes a esse mesmo parâmetro.
- É preciso marcar os diferentes pronomes que podem aparecer em uma sentença de modo a deixar claro se eles se referem ou não ao mesmo indivíduo.

Para distinguirmos as diferentes instâncias pronominais no interior de uma mesma sentença, vamos fazer uso de índices numéricos. Assim, nossas sentenças acima serão representadas da seguinte forma:

(18) Ele$_1$ é italiano.

(19) Ele$_1$ é tio dele$_2$.

Em (19), o uso de índices diferentes indica que os pronomes têm referentes distintos. Já na sentença a seguir, os dois pronomes têm o mesmo referente, já que são marcados como mesmo índice:

(20) O tio dele$_1$ deu um carro pra ele$_1$.

Para explicitarmos o papel do contexto e, com isso, especificarmos a extensão de um pronome, bem como as condições de verdade de uma sentença em que ele apareça, vamos nos valer de um conceito da lógica: a ATRIBUIÇÃO. Uma atribuição é uma função que leva números naturais em elementos de um certo domínio semântico. No nosso caso, esse domínio é D_e, o domínio dos indivíduos. Neste livro, vamos seguir uma implementação proposta em Heim & Kratzer (1998) e tratar as atribuições como FUNÇÕES PARCIAIS que têm como domínios subconjuntos dos números naturais. Todas as funções abaixo, por exemplo, são atribuições:

(21) a. $\left[1 \rightarrow \text{João}\right]$ b. $\begin{bmatrix} 1 \rightarrow \text{João} \\ 2 \rightarrow \text{Pedro} \end{bmatrix}$ c. $\begin{bmatrix} 1 \rightarrow \text{João} \\ 2 \rightarrow \text{Pedro} \\ 3 \rightarrow \text{Maria} \end{bmatrix}$

Atribuições, junto com os índices, nos permitem formalizar a variabilidade intrínseca das extensões dos pronomes. Assim, abre-se a possibilidade de captar

parte do papel do contexto de fala na determinação dessas extensões e das extensões dos constituintes que delas dependem. De maneira geral, e como é costume nos livros de lógica, iremos nos referir a atribuições pela letra *g* usada como um expoente do símbolo de extensão: $[\![\alpha]\!]^g$. A seguir, estão alguns exemplos de entradas lexicais para os pronomes:

(22) $[\![\text{ele}_1]\!]^g$ = g(1)

(23) $[\![\text{ele}_2]\!]^g$ = g(2)

(24) $[\![\text{ela}_3]\!]^g$ = g(3)

Generalizando, essas entradas lexicais devem ser lidas da seguinte forma: para qualquer atribuição *g*, e qualquer número natural *i*, a extensão de ele_i ou ela_i é o valor que *g* atribui a *i*, ou seja *g(i)*. Por exemplo, se *g* for a atribuição em (21c), então temos que:

(25) $[\![\text{ele}_1]\!]^g$ = João

(26) $[\![\text{ele}_2]\!]^g$ = Pedro

(27) $[\![\text{ela}_3]\!]^g$ = Maria

Já para sentenças contendo pronomes, o que teremos é o seguinte:

(28) $[\![\text{ele}_1 \text{ é italiano}]\!]^g$ = 1 *sse* João é italiano

(29) $[\![\text{ele}_2 \text{ é italiano}]\!]^g$ = 1 *sse* Pedro é italiano

(30) $[\![\text{ela}_3 \text{ gosta dele}_1]\!]^g$ = 1 *sse* Maria gosta de João

(31) $[\![\text{ela}_3 \text{ gosta dele}_2]\!]^g$ = 1 *sse* Maria gosta de Pedro

Mas voltemos agora ao nosso cenário inicial em que a sentença (16), repetida abaixo como (32), foi enunciada sem a presença de um indivíduo saliente no contexto.

(32) Ele é italiano.

Como notamos, tal uso é inadequado, ficando no ar a pergunta: a quem o pronome se refere? Continuando na esteira do que foi proposto em Heim & Kratzer (1998), podemos associar tais usos a casos em que o índice de um pronome não pertence ao domínio da atribuição. Para tornar isso mais explícito, façamos uma pequena modificação no nosso esquema correspondente à entrada lexical dos pronomes:

Entrada Lexical dos Pronomes
Para qualquer pronome *pron*, qualquer atribuição g, e qualquer número natural i,

$$[\![pron_i]\!]^g = \begin{cases} g(i) & \text{se } i \in \textit{Domínio de } g \\ indefinido & \text{se } i \notin \textit{Domínio de } g \end{cases}$$

Em palavras: a extensão de $pron_i$ em relação a g é o valor que g atribui a i, se i pertence ao domínio de g, e indefinida, se i não pertence ao domínio de g.

Para formalizar o caso especial em que simplesmente não há indivíduos salientes no contexto, podemos pensar em uma atribuição nula, que é uma função cujo domínio é o conjunto vazio. Representaremos essa atribuição por $\varnothing$. Temos então o seguinte:

$[\![\text{ele}_1]\!]^{[1 \to \text{João}]}$ = João
$[\![\text{ele}_1]\!]^{[1 \to \text{Pedro}]}$ = Pedro
$[\![\text{ele}_2]\!]^{[1 \to \text{João}]}$ = *indefinido*
$[\![\text{ele}_1]\!]^{\varnothing}$ = *indefinido*

No nível sentencial, temos o seguinte:

$[\![\text{Ele}_1 \text{ é italiano}]\!]^{[1 \to \text{João}]}$ = 1 *sse* João é italiano
$[\![\text{Ele}_1 \text{ é italiano}]\!]^{[1 \to \text{Pedro}]}$ = 1 *sse* Pedro é italiano
$[\![\text{Ele}_1 \text{ é italiano}]\!]^{\varnothing}$ = *indefinido*

Para casos envolvendo mais de um pronome, a sentença ficará sem valor de verdade sempre que um ou mais índices não pertencerem ao domínio da atribuição, correspondendo a situações de fala em que um número insuficiente de indivíduos se tornaram salientes:

$[\![\text{Ele}_1 \text{ é tio dele}_2]\!]^{\begin{bmatrix} 1 \to \text{João} \\ 2 \to \text{Pedro} \end{bmatrix}}$ = 1 *sse* João é tio de Pedro
$[\![\text{Ele}_1 \text{ é tio dele}_2]\!]^{[1 \to \text{João}]}$ = *indefinido*
$[\![\text{Ele}_1 \text{ é tio dele}_2]\!]^{\varnothing}$ = *indefinido*

Note que a entrada lexical dos pronomes não faz menção a seus traços gramaticais (gênero, número, pessoa). Isso, claro, não parece semanticamente correto.

Afinal de contas, *ele* e *ela* não parecem sinônimos. Um falante competente sabe que, diante de um homem, por exemplo, usa-se *ele*, ao passo, que diante de uma mulher, usa-se *ela*. Mas nem sempre há uma relação transparente entre morfologia e semântica. Uma sentença como *João chamou uma pessoa, mas ela não escutou* não fornece indícios sobre o referente do pronome *ela* ser um homem ou uma mulher. Nesse caso, é o gênero gramatical do antecedente do pronome (o sintagma nominal *uma pessoa*) que parece determinar a escolha da forma feminina *ela*. No que segue, continuaremos simplificando, ignorando a relação de gênero entre pronomes e seus antecedentes e/ou referentes.

Por fim, uma questão técnica. Nossa entrada lexical para os pronomes faz menção explícita a atribuições. Para que não só a extensão dos pronomes, mas também a dos constituintes que os contenham, sejam sensíveis a atribuições, precisamos reformular nossos princípios composicionais de modo a fazer com que constituintes herdem essa sensibilidade de seus constituintes imediatos. Nossos dois princípios composicionais, o princípio dos nós não ramificados e o princípio de aplicação funcional, passam a ter as seguintes versões:

Princípio dos nós não ramificados
Se α é um nó não ramificado cujo único constituinte imediato é β, então para qualquer atribuição g, $[\![\alpha]\!]^g = [\![\beta]\!]^g$.

Aplicação funcional
Seja α um nó ramificado, cujos constituintes imediatos são β e γ. Para qualquer atribuição g, se $[\![\beta]\!]^g$ é uma função e $[\![\gamma]\!]^g$ pertence ao domínio de $[\![\beta]\!]^g$, então $[\![\alpha]\!]^g = [\![\beta]\!]^g([\![\gamma]\!]^g)$.

Para que esses princípios se apliquem adequadamente, vamos assumir que as extensões de todos os itens lexicais são relativizadas a atribuições. Assim, para qualquer atribuição g, temos que:

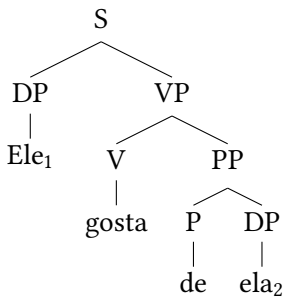
(33) a. $[\![\text{ele}_1]\!]^g = g(1)$
b. $[\![\text{João}]\!]^g = \textit{joão}$
c. $[\![\text{ama}]\!]^g = \lambda x.\ \lambda y.\ \textsc{ama}(y,x)$

Relativizar todas as entradas lexicais a atribuições pode parecer pôr em risco as distinções que salientamos acima entre pronomes e nomes próprios, por exemplo. Mas esse não é o caso. Note uma diferença crucial entre as extensões em

(33a) e (33b). Enquanto a primeira varia de atribuição para atribuição, a segunda não, já que a variável g não aparece do lado direito da igualdade. Ou seja, para qualquer atribuição g, a extensão do nome *João* (ou do verbo *amar* em (33c)) será a mesma. Com pronomes, entretanto, a situação é diferente, conforme discutimos acima. Por isso, pronomes são frequentemente chamados de VARIÁVEIS, em oposição a nomes próprios, chamados de CONSTANTES (termos emprestados da lógica de predicados). As distinções que discutimos anteriormente ficam, portanto, preservadas. Para efeitos de simplicidade, costuma-se omitir o expoente g quando a extensão em questão é insensível à natureza da atribuição. Esse é o caso de todos os nomes próprios, descrições definidas e predicados que havíamos analisado até aqui.

Vamos encerrar interpretando, passo a passo, uma sentença contendo dois pronomes, cada um com um índice diferente. Valeremo-nos de todo o material que introduzimos nesta seção. A estrutura sintática da sentença está representada na árvore logo a seguir. Assumiremos aqui que pronomes pessoais, como descrições definidas, são DPs. Os pronomes, entretanto, não tomam um NP complemento, sendo assim DPs não ramificados:

(34) Ele$_1$ gosta dela$_2$



1. $[\![\text{ela}_2]\!]^g = g(2)$	(L)
2. $[\![\text{PP}]\!]^g = [\![\text{DP}]\!]^g = [\![\text{ela}_2]\!]^g = g(2)$	(1, NR, *de* é vácua)
3. $[\![\text{gosta}]\!]^g = \lambda x.\lambda y.$ GOSTA(y,x)	(L)
4. $[\![\text{V}]\!]^g = [\![\text{gosta}]\!]^g = \lambda x.\lambda y.$ GOSTA(y,x)	(3, NR)
5. $[\![\text{VP}]\!]^g = [\![\text{V}]\!]^g([\![\text{PP}]\!]^g)$	(AF)
6. $[\![\text{VP}]\!]^g = (\lambda x.\lambda y.$ GOSTA(y,x))($g(2)$)	(2, 4, 5)
7. $[\![\text{VP}]\!]^g = \lambda y.$ GOSTA(y,*g(2)*)	(6,Conversão-λ)
8. $[\![\text{ele}_1]\!]^g = \text{g}(1)$	(L)
9. $[\![\text{DP}_{suj}]\!]^g = [\![\text{ele}_1]\!]^g = g(1)$	(8, NR)
10. $[\![\text{S}]\!]^g = [\![\text{VP}]\!]^g([\![\text{NP}_{suj}]\!]^g)$	(AF)

11. $[\![\text{S}]\!]$ = (λy. GOSTA(y,*g(2)*))(g(1)) (7, 9, 10)
12. $[\![\text{S}]\!]^g$ = 1 *sse* GOSTA(*g(1)*,*g(2)*) (11,Conversão-λ)

As condições de verdade de (34) dependem, portanto, do valor atribuído a g. Por exemplo,

$$[\![\text{S}]\!]^{\begin{bmatrix}1 \to \text{Pedro}\\ 2 \to \text{Maria}\end{bmatrix}} = 1 \textit{ sse } \text{Pedro gosta de Maria}$$

$$[\![\text{S}]\!]^{\begin{bmatrix}1 \to \text{João}\\ 2 \to \text{Marta}\end{bmatrix}} = 1 \textit{ sse } \text{João gosta de Marta}$$

$[\![\text{S}]\!]^{\varnothing}$ = *indefinido*

No caso da atribuição $\varnothing$, o sistema não confere valor de verdade à sentença. Isto deriva do fato de que as extensões dos pronomes são indefinidas, o que faz com que todos os constituintes que dominam esses pronomes, incluindo *S*, tenham suas extensões indefinidas. Isso, por sua vez, segue do fato de que nem o princípio dos nós não ramificados, nem aplicação funcional podem ser utilizados para a obtenção das extensões destes nós.

Terminamos assim nossa primeira incursão na semântica dos pronomes. Voltaremos a falar deles e, sobretudo, do conceito de atribuição nos próximos capítulos.

Sugestões de leitura

A semântica dos nomes próprios e das descrições definidas foi alvo da atenção de muitos filósofos, incluindo os notáveis e influentes Gottlob Frege, Bertrand Russell, P.F. Strawson, John Searle, Keith Donnellan, David Kaplan e Saul Kripke. Coletâneas reunindo os textos desses (e outros) autores sobre esses (e de outros) temas ligados à filosofia da linguagem incluem Ostertag (1998), Martinich (2001) e Ludlow (1998). Obras mais recentes sobre o tema incluem Abbott (2010), Elbourne (2013) e Neale (1990). Para uma apresentação acessível e abrangente sobre a semântica dos pronomes, ver Büring (2005). Para um excelente panorama sobre a interpretação das descrições definidas (e indefinidas), ver Heim (1991).

Exercícios

I. Calcular, passo a passo, as condições de verdade da sentença abaixo:

(1) A mãe do João adora ele$_3$.

II. Considere a seguinte sentença:

(1) Ele$_1$ é o João.

(A) Com base nas definições abaixo, e assumindo que a descrição definida *o João* é um designador rígido, mostre que não existe nenhuma atribuição g tal que (1) seja uma contingência em relação a g:

(D1) S é uma *tautologia* em relação a g se, e somente se, para toda situação possível, $[\![S]\!]^g = 1$.

(D2) S é uma *contradição* em relação a g se, e somente se, para toda situação possível, $[\![S]\!]^g = 0$.

(D3) S é uma *contingência* em relação a g se, e somente se, existir pelo menos uma situação possível tal que $[\![S]\!]^g = 0$ e pelo menos uma situação possível tal que $[\![S]\!]^g = 1$.

(B) Mostre que se trocarmos as definições (D1)-(D3) acima por (D1′)-(D3′) abaixo, a sentença (1) não será nem uma tautologia, nem uma contradição, e nem uma contingência.

(D1′) S é uma *tautologia* se, e somente se, para toda atribuição g e para toda situação possível, $[\![S]\!]^g = 1$.

(D2′) S é uma *contradição* se, e somente se, para toda atribuição g e para toda situação possível, $[\![S]\!]^g = 0$.

(D3′) S é uma *contingência* se, e somente se, para toda atribuição g, existir pelo menos uma situação possível tal que $[\![S]\!]^g = 0$ e pelo menos uma situação possível tal que $[\![S]\!]^g = 1$.

(C) Mostre que, se mantivermos (D1′) e (D2′) acima, mas trocarmos (D3′) por (D3″) abaixo, a sentença (1) será uma contingência:

(D3″) *S* é uma *contingência* se, e somente se, existir pelo menos uma atribuição g e uma situação possível tal que $[\![S]\!]^g = 0$ e pelo menos uma atribuição $g\prime$ e uma situação possível tal que $[\![S]\!]^{g'} = 1$.

(D) Para cada um dos três conjuntos de definições apresentados acima, diga se as sentenças abaixo são tautologias, contradições ou contingências, justificando sempre sua resposta:

(2) João nasceu no Brasil.

(3) Ele$_1$ nasceu no Brasil.

(4) João nasceu no Brasil e João não nasceu no Brasil.

(5) Ele$_1$ nasceu no Brasil e ele$_1$ não nasceu no Brasil.

(6) João nasceu no Brasil ou João não nasceu no Brasil.

(7) Ele$_1$ nasceu no Brasil ou ele$_1$ não nasceu no Brasil.

(8) João nasceu no Brasil e ele$_1$ não nasceu no Brasil.

(9) Ele$_1$ nasceu no Brasil e ele$_2$ não nasceu no Brasil.

5 Modificação nominal

Neste capítulo, discutiremos a interpretação de sintagmas nominais modificados por sintagmas adjetivais, sintagmas preposicionais e orações relativas. Do ponto de vista teórico, veremos como relacionar o uso dos sintagmas adjetivais e preposicionais como predicados principais e como modificadores nominais. Além disso, veremos como analisar semanticamente orações relativas a fim de que também possam servir como modificadores. Voltaremos a utilizar ATRIBUIÇÕES e introduziremos um novo princípio composicional, chamado de ABSTRAÇÃO FUNCIONAL.

5.1 Modificadores nominais

No capítulo 2, discutimos o uso de sintagmas nominais, adjetivais e preposicionais que funcionam como predicados principais em sentenças como as em (1) a seguir:

(1) a. João é médico.
    b. João é feliz.
    c. João é de Campinas.

Já no capítulo anterior, analisamos descrições definidas em que um sintagma nominal aparece junto ao artigo definido como na sentença em (2):

(2) O cachorro fugiu.

Vamos agora discutir alguns casos em que dois ou mais desses elementos se combinam, como nos exemplos abaixo:

(3) a. Sultão é um cachorro branco.
    b. João é um médico de Campinas.
    c. O cachorro branco de Campinas fugiu.

Nesses casos, diremos que os sintagmas adjetivais e preposicionais modificam os sintagmas nominais juntos dos quais aparecem. Em (3a), por exemplo, temos o nome *cachorro* modificado pelo adjetivo *branco* formando o predicado *cachorro branco*. Para que (3a) seja verdadeira, não basta apenas que Sultão seja um cachorro, ou apenas que ele seja branco, mas sim que ele seja as duas coisas, isto é, um cachorro branco. Da mesma forma, em (3b), João é descrito como um indivíduo que é tanto médico quanto proveniente de Campinas. Já em (3c), quem fugiu foi um animal que tem três características: é um cachorro, é branco e é de Campinas.

Além dos sintagmas adjetivais e preposicionais, analisaremos também o uso de orações relativas como modificadores nominais, como em (4):

(4) O cachorro que o João comprou fugiu.

Nesse caso, o referente em questão satisfaz duas condições: ser um cachorro e ter sido comprado pelo João. Orações relativas exercem um papel particularmente importante na modificação nominal. Essas orações fornecem um estoque potencialmente ilimitado de modificadores, sem que se precise alargar infinitamente o léxico com adjetivos e preposições relacionadas a critérios arbitrariamente específicos, como seria requerido, por exemplo, para replicar o sentido do DP abaixo:

(5) O cachorro que o João comprou de um chinês no início da semana passada durante uma feira de animais na cidade de São Paulo.

Sendo modificadores complexos construídos no nível sentencial, tira-se partido da composicionalidade e da recursividade típicas das línguas naturais, aliviando o papel do léxico.

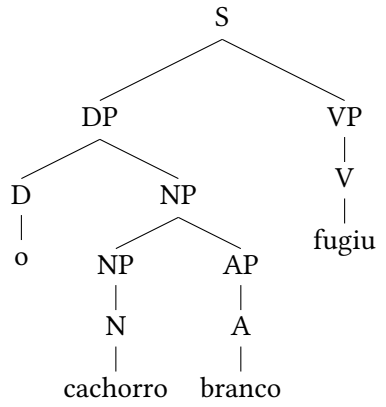
Note-se, por fim, que, além de orações relativas, os modificadores nominais podem também ser orações infinitivas (como em *documentos para assinar*) ou gerundivas (como em *caixas contendo medicamentos*). Esses, entretanto, não serão analisados semanticamente neste livro.

5.2 Modificação adjetival

Comecemos pela sentença abaixo:

(6) O cachorro branco fugiu.

Do ponto de vista sintático, vamos considerar o sintagma adjetival como um adjunto do sintagma nominal, conforme a representação abaixo:



Temos uma descrição definida na posição de sujeito encabeçada pelo artigo *o* e que toma por complemento um NP. Até aqui, nada de novo. A novidade é a natureza complexa desse NP, formado por um AP e outro NP. E, aqui, deparamonos com um problema. Quando analisamos NPs e APs no capítulo 2, tratamos a extensão desses sintagmas como sendo funções de tipo $\langle e, t\rangle$:

$[\![[_{\text{NP}}\text{ cachorro}]]\!] = \lambda x_e.\ \textsc{cachorro}(x)$
$[\![[_{\text{AP}}\text{ branco}]]\!] = \lambda x_e.\ \textsc{branco}(x)$

A questão agora é como obter a extensão do NP *cachorro branco*. Aplicação funcional não pode nos servir, já que temos duas funções de tipo $\langle e, t\rangle$, nenhuma podendo servir como argumento para a outra. Neste ponto, restam-nos duas alternativas. Na primeira, criamos um novo princípio composicional que especifique como obter a extensão de um sintagma cujos constituintes imediatos possuem, ambos, extensões de tipo $\langle e, t\rangle$. Na segunda, modificamos a extensão atribuída a NPs e/ou APs, de modo que a extensão de um possa servir de argumento à extensão do outro.

Vamos, primeiro, implementar essa segunda alternativa. Para não perdermos tudo o que vimos no capítulo anterior sobre a combinação de NPs com os determinantes *o*, *a*, vamos manter as extensões dos NPs como funções de tipo $\langle e, t\rangle$, e vamos alterar as extensões dos APs. Esqueçamos então, temporariamente, as extensões de tipo $\langle e, t\rangle$ que atribuímos a esses sintagmas.

Retornemos ao valor semântico do NP *cachorro branco*. Como as condições de

verdade de (6) nos dizem que quem fugiu deve ser não apenas um cachorro, mas também ser da cor branca, já sabemos o que queremos para a extensão do NP:

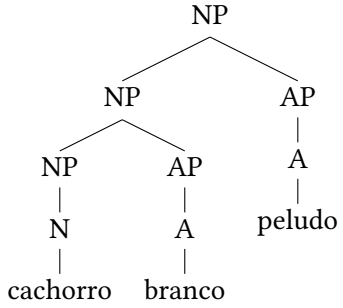
$[\![[_{\text{NP}}\text{ cachorro branco}]]\!] = \lambda x_e.\ \textsc{cachorro}(x)\ \&\ \textsc{branco}(x)$

Como tanto a extensão do NP *cachorro* quanto a do NP *cachorro branco* são de tipo $\langle e, t\rangle$, para valermo-nos de aplicação funcional, devemos atribuir à extensão de AP, e, por conseguinte, à extensão do adjetivo *branco*, o tipo $\langle\langle e, t\rangle, \langle e, t\rangle\rangle$. Informalmente, *branco* modifica a extensão do NP, retornando uma outra função que, por sua vez, retorna o valor 1 apenas para os indivíduos previamente caracterizados pela extensão do NP que são brancos. Pondo tudo isso, em termos formais, temos:

$[\![\text{branco}]\!] = \lambda F_{\langle e,t\rangle}.\lambda x_e.\ F'(x)\ \&\ \textsc{branco}(x)$

$[\![[_{\text{NP}}\text{ cachorro branco}]]\!] = [\![[_{\text{AP}}\text{ branco}]]\!]([\![[_{\text{NP}}\text{ cachorro}]]\!])$

Essa igualdade entre os tipos semânticos da extensão do NP que serve de argumento à extensão de AP e da extensão do NP resultante permite derivar a extensão de NPs modificados por mais de um adjetivo, como no exemplo a seguir, através de múltiplas instâncias de aplicação funcional:



$[\![[_{\text{NP}}\text{ cachorro branco peludo}]]\!] = [\![[_{\text{AP}}\text{ peludo}]]\!]([\![[_{\text{NP}}\text{ cachorro branco}]]\!])$

Como já sabemos que:

$[\![[_{\text{NP}}\text{ cachorro branco}]]\!] = [\![[_{\text{AP}}\text{ branco}]]\!]([\![[_{\text{NP}}\text{ cachorro}]]\!])$

Temos que:

$[\![[_{\text{NP}}$ cachorro branco peludo$]]\!] = [\![[_{\text{AP}}$ peludo$]]\!]([\![[_{\text{AP}}$ branco$]]\!]([\![[_{\text{NP}}$ cachorro$]]\!]))$

$[\![[_{\text{NP}}$ cachorro branco peludo$]]\!] = \lambda x_e.\ \textsc{cachorro}(x)\ \&\ \textsc{branco}(x)\ \&\ \textsc{peludo}(x)$

Temos, portanto, como interpretar estruturas em que um ou mais adjetivos modificam um sintagma nominal. Em todos esses casos, o resultado é uma função de tipo $\langle e, t\rangle$, que em casos como (6) é o tipo de argumento esperado pela extensão do artigo definido, conforme vimos no último capítulo. Resta-nos, agora, retornar aos casos em que o adjetivo não aparece junto a nenhum nome, como em *Sultão é branco*. Faremos isso na próxima seção, em que introduziremos uma nova ferramenta analítica.

5.3 Mudança de tipos

Consideremos o exemplo a seguir:

(7) Sultão é branco.

Para casos como esse, precisamos ainda de nossa antiga entrada lexical que atribuía à extensão de adjetivos como *branco* o tipo $\langle e, t\rangle$. Isto equivale a dizer que precisamos assumir que esse adjetivo é ambíguo e que no léxico estão listadas as duas entradas abaixo (vamos usar a marca $'$ para distinguir as entradas):

$[\![\text{branco}]\!] = \lambda x_e.\ \textsc{branco}(x)$

$[\![\text{branco}']\!] = \lambda F_{\langle e,t\rangle}.\ F'(x)\ \&\ \textsc{branco}(x)$

O mesmo valeria pra outros adjetivos, como *peludo*, por exemplo:

$[\![\text{peludo}]\!] = \lambda x_e.\ \textsc{peludo}(x)$

$[\![\text{peludo}']\!] = \lambda F_{\langle e,t\rangle}.\ F'(x)\ \&\ \textsc{peludo}(x)$

Duplicando, uma a uma, as entradas lexicais de todos os adjetivos, deixamos escapar a generalização de que a ambiguidade de tipos $\langle e, t\rangle$ e $\langle\langle e, t\rangle, \langle e, t\rangle\rangle$ não é uma característica acidental encontrada nesse ou naquele adjetivo, mas um fenômeno geral que parece afetar toda a classe dos adjetivos. Além disso, nossa intuição parece clara em reconhecer que a semelhança entre os adjetivos nos pares de sentenças abaixo não é apenas fonológica, mas também semântica. Ou

seja, os usos dos adjetivos em cada um dos membros desses pares nos parecem claramente relacionados do ponto de vista do significado. Parece justo, portanto, cobrar de uma teoria semântica a explicitação dessa relação.

Por fim, cumpre notar que não são apenas adjetivos listados no léxico, mas também sintagmas adjetivais e preposicionais complexos, como *fiel ao João* ou *de Campinas*, que podem aparecer acompanhando ou não um sintagma nominal, como atestam os exemplos abaixo:

(8) a. Sultão é fiel ao João.
b. O cachorro fiel ao João fugiu.

(9) a. João é de Campinas.
b. O médico de Campinas chegou.

Para estes casos, também precisaríamos ter duas possibilidades:

$[\![[_{\text{AP}}$ fiel ao João$]]\!]$ = λx_e. FIEL(*x*,*joão*)
$[\![[_{\text{AP}}$ fiel ao João$]]\!]$ = $\lambda F_{\langle e,t\rangle}$. λx_e. $F'(x)$ & FIEL(*x*,*joão*)

$[\![[_{\text{PP}}$ de Campinas$]]\!]$ = λx_e. SER_DE(*x*,*campinas*)

$[\![[_{\text{PP}}$ de Campinas$]]\!]$ = $\lambda F_{\langle e,t\rangle}$. λx_e. $F'(x)$ & SER_DE(*x*,*campinas*)

Para captar essa generalização acerca da possibilidade de uso de sintagmas adjetivais e preposicionais acompanhados ou não de um sintagma nominal, vamos formular uma regra que se aplica a todos os sintagmas adjetivais, relacionando formalmente as duas extensões que se podem atribuir a eles. Trata-se de uma REGRA DE MUDANÇA DE TIPOS, que assume que o tipo básico das extensões dos APs é $\langle e,t\rangle$ e explicita como obter uma extensão de tipo $\langle\langle e,t\rangle,\langle e,t\rangle\rangle$ a partir da extensão de tipo $\langle e,t\rangle$:

Regra de Mudança de Tipos para Sintagmas Adjetivais e Preposicionais
Seja X um sintagma adjetival ou preposicional cuja extensão α é de tipo $\langle e,t\rangle$. Mude a extensão de X de α para α', sendo α' de tipo $\langle\langle e,t\rangle,\langle e,t\rangle\rangle$ e definida da seguinte forma:

α' = $\lambda F_{\langle e,t\rangle}.\lambda x_e$. $F'(x)$ & $\alpha(x) = 1$

A aplicação desta regra fica condicionada ao contexto sintático em que o sintagma aparecer. Assim, para o adjetivo *branco*, por exemplo, basta listá-lo no léxico como denotando uma função de tipo $\langle e, t\rangle$. Isso (junto com o princípio dos nós não ramificados) nos daria a extensão do AP necessária para a interpretação de sentenças como (7):

$[\![\text{AP}]\!] = \lambda x.\ \textsc{branco}(x)$

Para sentenças como (6), em que o AP modifica um NP, recorreríamos à regra acima, que nos forneceria o seguinte:

$[\![\text{AP}]\!] = \lambda F_{\langle e,t\rangle}.\lambda x_e.\ F'(x)\ \&\ [\![\text{AP}]\!](x) = 1$

$[\![\text{AP}]\!] = \lambda F_{\langle e,t\rangle}.\lambda x_e.\ F'(x)\ \&\ (\lambda x.\ \textsc{branco}(x))(x) = 1$

$[\![\text{AP}]\!] = \lambda F_{\langle e,t\rangle}.\lambda x_e.\ F'(x)\ \&\ \textsc{branco}(x)$

O leitor poderá verificar a adequação dessa regra, aplicando-a na derivação das condições de verdade das demais sentenças apresentadas nesta seção (ver exercício I). O recurso a uma regra de mudança de tipos nos proporcionou captar a flexibilidade atrelada ao uso dos sintagmas adjetivais e preposicionais sem apelarmos a ambiguidades idiossincráticas vinculadas separadamente a cada adjetivo listado no léxico. O preço pago foi a postulação de uma nova regra semântica. Na próxima seção, veremos uma alternativa que mantém uma única interpretação para os APs e PPs. Mas já adiantamos que também haverá um preço a ser pago.

5.4 Conjunção funcional

Voltemos à tarefa a que nos propusemos no início do capítulo: interpretar expressões como *cachorro branco*, sabendo que seus constituintes imediatos tinham como extensões funções de tipo $\langle e, t\rangle$. Desta vez, vamos analisar um tratamento alternativo, ao qual fizemos alusão mais acima, em que os tipos semânticos de NPs e APs permanecem inalterados, mas um novo princípio composicional é introduzido. Chamemos este novo princípio de Conjunção funcional:

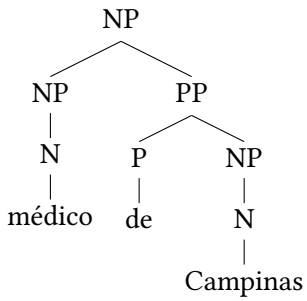
Conjunção funcional
Seja α um nó ramificado, cujos constituintes imediatos são β e γ, tal que $[\![\beta]\!]$ e $[\![\gamma]\!]$ pertençam a $D_{\langle e,t\rangle}$. Neste caso, $[\![\alpha]\!] = \lambda x_e.\ [\![\beta]\!](x) = 1$ & $[\![\gamma]\!](x) = 1$

Conjunção funcional permite que obtenhamos uma extensão de tipo $\langle e, t\rangle$ para um nó cujos constituintes imediatos possuem, ambos, extensões do tipo $\langle e, t\rangle$. Note a semelhança com o que vimos no capítulo 3 sobre a interpretação da conjunção *e* em casos como [$_{VP}$ fuma e bebe] ou NPs como [$_{NP}$ médico e psicólogo], inclusive no uso do conectivo lógico & na metalinguagem. Voltando a nosso exemplo, utilizamos o nosso novo princípio para obter a extensão do NP *cachorro branco* da seguinte forma:

$[\![[_{NP}$ cachorro branco$]]\!] = \lambda x_e.\ [\![[_{NP}$ cachorro$]]\!](x) = 1$ & $[\![[_{AP}$ branco$]]\!](x) = 1$

$[\![[_{NP}$ cachorro branco$]]\!] = \lambda x_e.$ CACHORRO(x) & BRANCO(x)

Conjunção funcional também nos serve nos casos em que um PP modifica um NP, como em *médico de Campinas*:



Note que este uso da preposição *de* indicando origem geográfica difere do uso que vimos quando tratamos de NPs como *irmão de Maria* ou APs como *fiel ao João*. Para esses últimos, a preposição foi tratada como um item semanticamente vácuo. Para o caso acima, o que queremos é o seguinte:

$[\![\text{de}]\!] = \lambda x_e.\ \lambda y_e.$ SER_DE(y,x)

Como o leitor já pode antecipar, a extensão do PP *de Campinas* é obtida através de Aplicação Funcional. Já a extensão do NP *médico de Campinas* é obtida utilizando-se Conjunção funcional:

$[\![\text{de Campinas}]\!] = [\![\text{de}]\!]([\![\text{Campinas}]\!]) = \lambda y.\ \textsc{ser_de}(y,\textit{campinas})$
$[\![\text{médico}]\!] = \lambda y.\ \textsc{médico}(y)$
$[\![\text{médico de Campinas}]\!] = \lambda y.\ [\![\text{médico}]\!] = 1\ \&\ [\![\text{de Campinas}]\!] = 1$
$[\![\text{médico de Campinas}]\!] = \lambda y.\ \textsc{médico}(y)\ \&\ \textsc{ser_de}(y,\textit{campinas})$

No restante deste capítulo, continuaremos a nos valer da regra de mudança de tipos formulada mais acima em vez da conjunção funcional. Tal escolha, entretanto, é de certo modo arbitrária, não estando baseada em nenhuma consideração teórica ou empírica. Tanto em um caso como no outro, aplicação funcional precisará ser acompanhada de de uma nova regra para que todos os casos relevantes sejam cobertos.

5.5 Modificação e classes de comparação

Considere o argumento abaixo:

(10) Sultão é um cachorro branco.
Todo cachorro é um animal.
Logo, Sultão é um animal branco.

Trata-se de um argumento válido, ou seja, um argumento em que a verdade das premissas leva necessariamente à verdade da conclusão. Nosso sistema nos ajuda a entender esse fato. Das condições de verdade que atribuímos à primeira premissa, sabemos que se ela é verdadeira, então Sultão é um cachorro e Sultão é branco. A verdade da segunda premissa (cuja interpretação discutiremos no próximo capítulo) nos diz que, para todo indivíduo, se esse indivíduo é um cachorro, então ele é um animal. Como já sabemos que Sultão é um cachorro, logo concluímos que Sultão é um animal. Como da verdade da primeira premissa também já sabemos que Sultão é branco, concluímos que Sultão é um animal e Sultão é branco. E isso satisfaz as condições de verdade da conclusão do argumento.

Considere agora o seguinte argumento:

(11) Lili é uma formiga grande.
Toda formiga é um animal.
Logo, Lili é um animal grande.

Esse argumento não parece válido. Ainda que assumamos a verdade das premissas, a verdade da conclusão não se segue. A razão é que uma formiga grande pode muito bem ser um animal pequeno. Ou seja, quando usamos o adjetivo *grande* na primeira premissa, estamos muito provavelmente comparando o tamanho de Lili com algo como o tamanho médio das formigas. Já no caso da conclusão, parecemos estar comparando o tamanho de Lili com o tamanho médio dos animais em geral, ou, pelo menos, de alguns animais mais comuns como formigas, cachorros, tigres, elefantes, etc...Diferentemente do significado do adjetivo *branco*, o significado de adjetivos como *grande* parece fazer menção a uma classe de comparação, a partir da qual se estabelece um parâmetro de avaliação para se dizer se o adjetivo se aplica ou não a um determinado indivíduo. Como vimos acima, a natureza do nome que o adjetivo modifica influencia a escolha dessa classe: a classe das formigas, no caso do NP *formiga grande*, ou a classe dos animais, no caso do NP *animal grande*. Mas não é só o contexto linguístico que influencia a escolha da classe de comparação. Pense no uso do adjetivo *alto* na sentença abaixo:

(12) João é alto.

Note primeiro que o adjetivo não aparece adjacente a qualquer substantivo comum. Imagine agora que João meça um metro e oitenta e cinco centímetros, e que alguém nos pergunte se (12) é verdadeira ou falsa. Diferentes contextos levariam a diferentes respostas. Por exemplo, se João é uma criança de dez anos, certamente diremos que (12) é verdadeira. Se João já é um adulto jogador de basquete, já não estaríamos tão certos, e provavelmente acrescentaríamos que *pra um jogador de basquete*, João não é alto. Nota-se, portanto, que a classe de comparação em questão pode vir também do contexto extrasentencial e que, à medida que uma conversa segue seu curso, diferentes classes de comparação podem ser usadas na interpretação de adjetivos como *grande* ou *alto*.

Para captarmos essa dependência contextual e para bloquearmos inferências como a do argumento em (11), precisamos tratar adjetivos como *grande* ou *alto* de maneira distinta da que tratamos *branco*. Não iremos de modo algum propor uma teoria detalhada sobre o estabelecimento e a natureza de uma classe de comparação, muito menos elucidar questões profundas sobre dependência contextual. Nosso objetivo aqui é mais modesto. Queremos apenas indicar na entrada lexical de certos adjetivos a influência do contexto, ainda que de maneira rudimentar. Assim, contentemo-nos com o seguinte:

$[\![\text{grande}]\!] = \lambda x_e$. o tamanho de x está acima do padrão estabelecido pela classe de comparação fornecida pelo contexto de fala.

A não validade do argumento em (11) decorre, portanto, do fato de que o contexto de fala muda entre o momento que falamos/ouvimos a primeira premissa e o momento que falamos/ouvimos a conclusão, tornando a extensão da ocorrência do adjetivo *grande* na primeira premissa diferente da extensão da ocorrência do mesmo na conclusão.

A despeito das diferenças apontadas acima entre o comportamento lógico-semântico de *branco* e *grande*, sua aplicação a um NP guarda uma semelhança. Em ambos os casos, sentenças da forma *DP é um NP AP* acarretam sentenças da forma *DP é um NP*. Concretamente, se Sultão é um cahorro branco, então Sultão é um cachorro. Igualmente, se Sultão é um cachorro grande, então Sultão é um cachorro. Gostaríamos, antes de finalizar essa seção, de mencionar que há adjetivos para os quais essas inferências não são válidas e que colocam problemas adicionais ao nosso sistema, seja ele baseado na ideia de modificação como aplicação funcional ou baseado no princípio de conjunção funcional. Pense, por exemplo, em adjetivos como *suposto*. Dizer que alguém é um suposto criminoso não nos leva à conclusão de que esse alguém é um criminoso. Ou pense em adjetivos como *falso*. Seria um diamante falso um diamante? E um relógio falso, seria um relógio? Muito provavelmente, sua reposta aqui seria um inseguro 'talvez'. Esses casos, que não discutiremos aqui, desafiam o tratamento que demos à modificação adjetival e mesmo o projeto de uma semântica extensional, já que a extensão de NPs como *suposto criminoso* ou *diamante falso* não parecem sequer depender da extensão dos respectivos substantivos modificados (sobre isso, ver algumas das sugestões de leitura ao final do capítulo).

5.6 Orações relativas

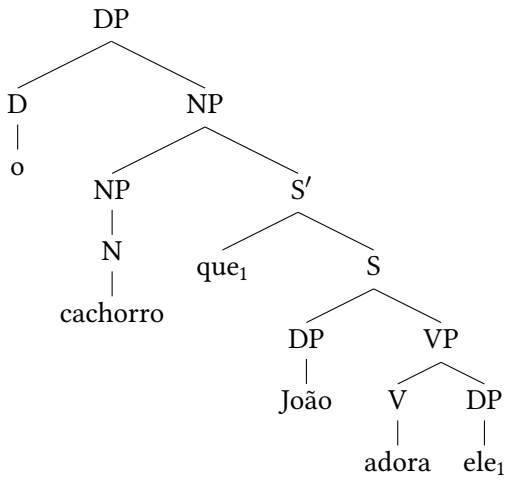
Como já mencionamos na introdução deste capítulo, assim como sintagmas adjetivais e preposicionais podem modificar sintagmas nominais, também constituintes sentencias podem servir de modificadores nominais, na forma de orações relativas, por exemplo. Em (13) abaixo, estão alguns exemplos contendo variedades dessas orações encontradas em português:

(13) a. O cachorro que João adora fugiu.
b. O cachorro de que João gosta fugiu.
c. O cachorro que João gosta fugiu.
d. O cachorro que João adora ele fugiu.

Em (13a), temos uma oração relativa modificando o nome *cachorro*. A posição relativizada corresponde à posição de objeto direto do verbo *adorar*. O mesmo se dá em (13b)-(13c), exceto pelo fato de que o verbo *gostar* é transitivo indireto, requerendo o uso da preposição *de*. (13b) pertence ao registro formal da língua, enquanto (13c) e (13d) pertencem ao registro falado, usado com enorme frequência no português brasileiro falado. Em (13a)-(13c), a posição relativizada aparece vazia, enquanto em (13d), um pronome aparece nesta posição. Este pronome — *ele*, no caso — é chamado de pronome resumptivo.

Ocuparemo-nos aqui apenas de (13a) e (13d), já que (13b) e (13c) exigem um refinamento sintático que está além do que podemos oferecer neste livro. A esse respeito, cumpre dizer que, mesmo nos casos de (13a) e (13d), vamos nos esquivar de uma série de detalhes morfossintáticos que têm sido objeto de estudo de muitos sintaticistas e para os quais diversas análises alternativas têm sido propostas, não só para o português brasileiro, mas também para outras línguas.

Vamos começar por (13d), cujo sujeito receberá a seguinte estrutura:



Trataremos a palavra *que* que aparece no início da oração relativa como um pronome relativo e atribuiremos a ele o mesmo índice atribuído ao pronome resumptivo *ele* que aparece na posição relativizada, indicando que esses dois elementos estão relacionados. A oração relativa como um todo está adjungida ao NP *cachorro* e o NP resultante *cachorro que o João adora* atua como complemento do determinante *o*, formando um DP.

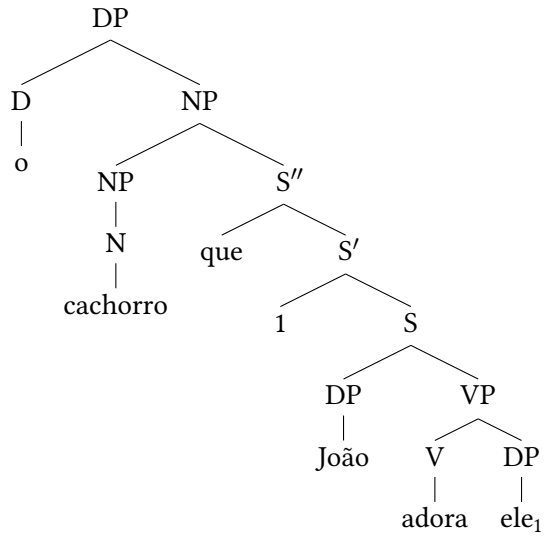
Passemos então à interpretação deste DP, começando pelo NP *cachorro que o João adora*. Façamos um paralelo com o que vimos anteriormente a respeito do

NP *cachorro branco*. Tratamos este NP como denotando uma função de tipo $\langle e, t\rangle$ que leva um indivíduo qualquer no valor de verdade 1, se, e somente se, este indivíduo for um cachorro e for também branco. Parece, então, natural tratar o NP *cachorro que o João adora* como denotando uma função de tipo $\langle e, t\rangle$ que leva um indivíduo qualquer no valor de verdade 1 se, e somente se, este indivíduo for um cachorro e for também um indivíduo que o João adora.

$[\![\text{cachorro que João adora}]\!] = \lambda x.\ \textsc{cachorro}(x)\ \&\ \textsc{adora}(\textit{joão},x)$

Para nossa oração relativa, queremos associá-la semanticamente a uma função de tipo $\langle e, t\rangle$ que leve um indivíduo x no valor de verdade 1 se, e somente se, João adora x. Dessa forma, poderemos estender nossa análise de modificação envolvendo sintagmas adjetivais e preposicionais para casos envolvendo orações relativas. A tarefa que nos resta, claro, é obter tal resultado composicionalmente.

Com esse intuito, vamos primeiro operar uma pequena modificação na estrutura que adotamos acima para a oração relativa. Seguindo uma proposta elaborada em Heim & Kratzer (1998), vamos assumir que, na interface sintaxe-semântica, o índice que atribuímos ao pronome relativo seja transferido do pronome para o constituinte irmão deste pronome, adjungindo-se a ele, como exemplificado abaixo:



Esse processo de transferência de índices é apenas uma conveniência, mas que será especialmente útil quando analisarmos estruturas derivadas por movimento

e a interpretação de sintagmas quantificadores no próximo capítulo.

Vejamos, então, como podemos proceder. Com relação ao constituinte S, já sabemos o que o nosso sistema deriva. Para uma atribuição g qualquer, temos:

$[\![\text{S}]\!]^g$ = 1 *sse* João adora g(1).

Como este constituinte contém um pronome, sua extensão depende da atribuição g. Mas note que, ao passarmos de *S* para a oração relativa completa, não queremos que esse pronome seja tratado dêitica ou anaforicamente. Ou seja, não queremos que ele se refira a um indivíduo tornado saliente pelo contexto de fala. Ao contrário, pela interpretação da oração relativa que descrevemos acima, esse pronome funciona como uma VARIÁVEL LIGADA, não necessitando se referir a um indivíduo contextualmente saliente.

Formalmente, para derivarmos a extensão que desejamos, precisamos de um mecanismo que nos permita obter uma função (tipo $\langle e, t\rangle$, no nosso caso) a partir da extensão de *S* (que é dependente de uma atribuição), levando-se em conta o índice do pronome relativo que marca a posição relativizada no interior de *S*. Para interpretar constituintes como S', que dominam um índice numérico, elaboraremos um novo princípio composicional, que chamaremos de **abstração funcional**. Vamos enunciá-lo e depois explicá-lo:

Abstração funcional
Seja α um nó ramificado cujos constituintes imediatos são um índice numérico *i* e β. Então, $[\![\alpha]\!]^g = \lambda x_e.[\![\beta]\!]^{g[i\rightarrow x]}$

Se você se lembrar do que vimos no capítulo 2 sobre a notação lambda, você perceberá que a expressão correspondente à extensão de α na regra acima é uma *abstração-*λ obtida a partir da extensão de β. Abstrações lambda, como já sabemos, representam funções. Daí o nome desse princípio. Quanto à notação $\underline{[\![\beta]\!]^{g[i\rightarrow x]}}$, ela deve ser interpretada da seguinte forma: a extensão de β em relação a uma atribuição g', que é idêntica a g exceto pelo fato de que g' mapeia o número i no indivíduo x, ou seja, $g'(i) = x$. Os exemplos abaixo ilustram a notação $g[i \rightarrow x]$:

$$g : \begin{bmatrix} 1 \rightarrow \text{João} \\ 2 \rightarrow \text{Pedro} \\ 3 \rightarrow \text{Maria} \end{bmatrix} \qquad g[1 \rightarrow \text{Carlos}] : \begin{bmatrix} 1 \rightarrow \text{Carlos} \\ 2 \rightarrow \text{Pedro} \\ 3 \rightarrow \text{Maria} \end{bmatrix}$$

$$g: \begin{bmatrix} 1 \rightarrow \text{João} \\ 2 \rightarrow \text{Pedro} \\ 3 \rightarrow \text{Maria} \end{bmatrix} \qquad g[3 \rightarrow \text{Paula}]: \begin{bmatrix} 1 \rightarrow \text{João} \\ 2 \rightarrow \text{Pedro} \\ 3 \rightarrow \text{Paula} \end{bmatrix}$$

$$g: \begin{bmatrix} 1 \rightarrow \text{João} \\ 2 \rightarrow \text{Pedro} \end{bmatrix} \qquad g[3 \rightarrow \text{Maria}]: \begin{bmatrix} 1 \rightarrow \text{João} \\ 2 \rightarrow \text{Pedro} \\ 3 \rightarrow \text{Maria} \end{bmatrix}$$

Note que para qualquer atribuição g, o domínio de $g[i \rightarrow x]$ será idêntico ao domínio de g se o número i já pertencer ao domínio de g. Entretanto, o domínio de $g[i \rightarrow x]$ incluirá o domínio de g se o número i não pertencer ao domínio de g. Mas o mais importante nesse ponto é notar que, para uma atribuição g, um indivíduo x e um índice i quaisquer, teremos sempre:

$$g[i \rightarrow x](i) = x$$

Antes de prosseguir, uma última e breve observação notacional. Adotamos aqui para as atribuições modificadas a notação $g[i \rightarrow x]$, que tomamos emprestada de Büring (2005), e que nos parece mais transparente e amigável do que a notação mais comum em manuais de lógica, que é $g[i/x]$. De qualquer forma, trata-se apenas de uma questão de gosto, sendo ambas equivalentes semanticamente.

Com essas definições e exemplos em mente, podemos usar a abstração funcional para a obtenção da extensão do constituinte S' na estrutura que estamos interpretando:

$$[\![S']\!]^g = \lambda x_e . [\![\text{S}]\!]^{g[1 \rightarrow x]}$$

Note que a mesma variável x que aparece junto ao operador lambda aparece na especificação da atribuição em relação à qual a extensão de S está relativizada. É isso que permitirá ao(s) pronome(s) que estivere(m) marcado(s) com o índice 1 no interior de S se comportarem semanticamente como variáveis ligadas. Vejamos os detalhes. S é uma sentença simples formada por um verbo transitivo e seus dois argumentos. Sua extensão, portanto, é obtida através de aplicação funcional (além, claro, do princípio dos nós não ramificados):

$$[\![S']\!]^g = \lambda x. \, [\![\text{adora}]\!]^{g[1 \rightarrow x]}([\![\text{ele}_1]\!]^{g[1 \rightarrow x]})([\![\text{João}]\!]^{g[1 \rightarrow x]})$$

As extensões do verbo *adorar* e do sujeito *João*, por não conterem pronomes, não

apresentam nada de novo:

$[\![\text{adora}]\!]^{g[1\to x]} = \lambda x.\lambda y.\ \textsc{adora}(y,x)$
$[\![\text{João}]\!]^{g[1\to x]} = \textit{joão}$

Em relação ao pronome *ele*, dado o que sabemos desde o capítulo anterior e o que acabamos de ver acima, temos:

$[\![\text{ele}_1]\!]^{g[1\to x]} = g[1 \to x](1) = x$

Juntando tudo isso, temos:

$[\![\text{S}']\!]^{g} = \lambda x.\ \textsc{adora}(\textit{joão},x)$

No fim das contas, o pronome resumptivo ele_1 acabou sendo interpretado como uma variável ligada pelo operador lambda. Esse, por sua vez, foi introduzido pela regra de abstração funcional ao se deparar com o índice 1, o mesmo índice do pronome. Isso foi possível pois aplicação funcional e o princípio dos nós não ramificados não interferem na atribuição, apenas a "passam adiante", por assim dizer. Isso permite uma conexão semântica entre o índice numérico que aparece junto ao pronome relativo no início da oração e os pronomes marcados com esse mesmo índice em seu interior, mesmo que haja uma distância sintática entre eles.

Voltando à nossa derivação, obtivemos o resultado que desejávamos, com o constituinte S′ se assemelhando semanticamente a um predicado de indivíduos. Note ainda que a extensão de S′ não depende da atribuição g, sendo sempre a mesma função que leva um indivíduo no valor 1 se, e somente se, João adora esse indivíduo. Isto é desejável, já que a contribuição da oração relativa que estamos analisando não parece mesmo depender do contexto.

Para obtermos a extensão do NP *cachorro que João adora*, temos duas opções, de acordo com o que vimos nas seções anteriores. A primeira delas é assumir que o pronome relativo seja semanticamente vácuo e nos valer do princípio de conjunção funcional, já que tanto a extensão do NP *cachorro* quanto a extensão da oração relativa *que o João adora ele* seriam de tipo $\langle e, t\rangle$. A segunda opção seria atribuir uma extensão ao pronome relativo e valermo-nos de aplicação funcional. Para tanto, essa extensão deve ser uma função de tipo $\langle et, \langle et, et\rangle\rangle$, definida da seguinte forma:

$[\![\text{que}_{\text{REL}}]\!]^{g} = \lambda F_{\langle e,t\rangle}.\lambda G_{\langle e,t\rangle}.\lambda x_e.\ G'(x)\ \&\ F'(x)$

Note que, de acordo com essa entrada, cabe ao pronome relativo o papel da conjunção dos predicados correspondentes à oração relativa e ao NP que ela modifica. Para a oração relativa S′′ acima, teríamos:

$[\![\text{S}'']\!]^g = [\![\text{que}_{\text{REL}}]\!]^g([\![\text{S}']\!]^g)$
$[\![S'']\!]^g = \lambda G_{\langle e,t\rangle}.\lambda x_e.\ G'(x)$ & ADORA(*joão,x*)

Note que essa extensão da oração relativa é de tipo $\langle et, et\rangle$, o mesmo tipo dos sintagmas adjetivais que modificam NPs, refletindo o caráter de modificador nominal destas orações. Como a extensão de *cachorro* é uma função de tipo $\langle e, t\rangle$, valemo-nos de aplicação funcional mais uma vez para chegarmos à extensão do NP *cachorro que o João adora ele*:

$[\![\text{NP}]\!]^g = [\![\text{S}'']\!]^g([\![\text{cachorro}]\!]^g)$
$[\![\text{NP}]\!]^g = \lambda x_e.$ CACHORRO(x) & ADORA(*joão,x*)

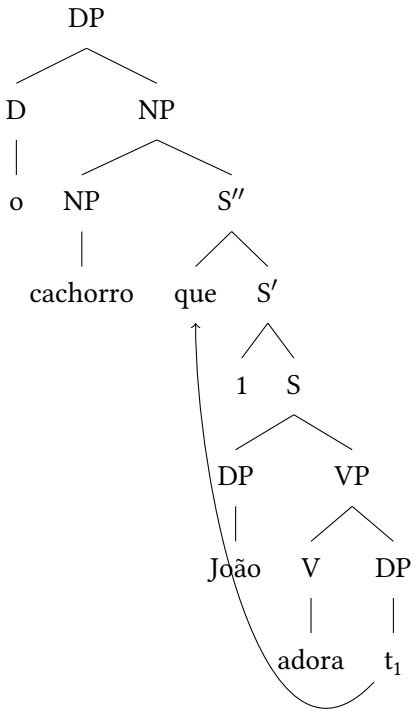
Por fim, para a obtenção da extensão do DP *o cachorro que o João adora ele*, basta aplicarmos a extensão do artigo definido à extensão do NP que acabamos de derivar e o resultado, como o leitor poderá verificar, será o seguinte: o único indivíduo *x* saliente no contexto, tal que *x* é um cachorro e o João adora *x*.

5.6.1 Vestígios e movimento sintático

A oração relativa que analisamos acima continha um pronome resumptivo na posição relativizada. Como já mencionamos, esse não é um traço comum a todas as orações relativas, podendo a posição relativizada aparecer vazia, como no exemplo (13a), repetido abaixo como (14):

(14) O cachorro que João adora fugiu.

Do ponto de vista sintático, analisaremos esse tipo de oração relativa de acordo com a Teoria da Regência e Ligação (ver referências ao final do capítulo). Dessa perspectiva teórica, o pronome relativo *que* se moveria da posição relativizada (objeto direto, no exemplo acima) até o início da oração, deixando em sua posição de partida um vestígio *t* com o mesmo índice e sem realização fonética. Essa análise está representada abaixo (assumindo o mesmo processo de transferência de índice discutido anteriormente):



Será essa estrutura resultante do movimento que o componente semântico interpretará. Para tanto, precisamos atribuir uma extensão ao vestígio indexado. Assumiremos aqui que orações relativas com e sem pronomes resumptivos contribuem de maneira idêntica para as condições de verdade das sentenças que as contêm e analisaremos os vestígios da mesma forma como analisamos os pronomes resumptivos, ou seja, como variáveis ligadas. Vestígios, ao contrário dos pronomes, não admitem interpretações dêiticas. Para nós, entretanto, isso não será relevante, já que admitiremos tratar-se de uma restrição puramente sintática. Temos, então, o seguinte:

Entrada Lexical dos Vestígios
Para qualquer vestígio *t*, atribuição *g* e número natural *i*,

$$[\![t_i]\!]^g = \begin{cases} g(i) & \text{se } i \in \text{Domínio}(g) \\ indefinido & \text{se } i \notin \text{Domínio}(g) \end{cases}$$

Dada a semelhança na interpretação de pronomes e vestígios, a derivação de (14) prossegue de maneira idêntica à derivação de sua contraparte com o pronome resumptivo. Consequentemente, as mesmas condições de verdade serão derivadas.

Nos exemplos que discutimos acima, os pronomes relativos estavam coindexados ou com um vestígio ou com um pronome. É possível, entretanto, que em uma mesma oração relativa haja um vestígio e um pronome, ambos coindexados com o mesmo pronome relativo. Considere o exemplo abaixo:

(15) O cachorro que mordeu o dono dele fugiu.

Em uma das leituras possíveis para (15), o pronome *ele* é interpretado deiticamente. Imagine, por exemplo, que estamos apontando para *Sultão*, um cachorro cujo dono foi mordido por um outro cachorro, que fugiu logo após a mordida. Essa leitura pode ser captada atribuindo ao pronome *ele* um índice diferente do índice atribuído ao pronome relativo e ao vestígio deixado na posição de sujeito:

(16) [cachorro [que$_1$ [t$_1$ mordeu o dono dele$_2$]]]

Essa atribuição de índices leva à seguinte extensão para a oração relativa:

$[\![\text{que}_1\ \text{t}_1\ \text{mordeu o dono dele}_2]\!]^g = \lambda x.\ \textsc{mordeu}(x, \textit{o dono de g(2)})$

Note que essa extensão depende do valor da atribuição *g*. Se, por exemplo, *g*(2) = Sultão, então teremos uma função que leva um indivíduo no valor de verdade 1 se, e somente se, este indivíduo tiver mordido o dono de Sultão. Vejamos, como exercício, a derivação dessa extensão. Após a transferência do índice do pronome relativo, temos a seguinte estrutura para a oração relativa:

(17) [$_{S''}$ que [$_{S'}$ 1 [$_S$ t$_1$ mordeu o dono dele$_2$]]]

Apresentamos abaixo a derivação, passo a passo, dessa estrutura para que o leitor possa conferir em detalhe a aplicação de tudo o que vimos até aqui. Procederemos de cima para baixo na estrutura:

1. $[\![\text{S}'']\!]^g = [\![\text{que}]\!]^g([\![\text{S}']\!]^g)$ (AF)

2. $[\![\text{S}']\!]^g = \lambda x.\ [\![\text{S}]\!]^{g[1\to x]}$ (AbF)

3. $[\![\text{S}'']\!]^g = [\![\text{que}]\!]^g(\lambda x.\ [\![\text{S}]\!]^{g[1\to x]})$ (1,2)

4. $[\![\text{S}]\!]^{g[1\to x]} = [\![\text{mordeu o dono dele}_2]\!]^{g[1\to x]}([\![\text{t}_1]\!]^{g[1\to x]})$ (AF)

5. $[\![\text{mordeu o dono dele}_2]\!]^{g[1\to x]} = [\![\text{mordeu}]\!]^{g[1\to x]}([\![\text{o dono dele}_2]\!]^{g[1\to x]})$ (AF)

6. $[\![\text{mordeu}]\!]^{g[1\to x]} = \lambda x.\lambda y.\ \textsc{mordeu}(y,x)$ (L)

7. $[\![\text{o dono dele}_2]\!]^{g[1\to x]} = [\![\text{o}]\!]^{g[1\to x]}([\![\text{dono dele}_2]\!]^{g[1\to x]})$ (AF)

8. $[\![\text{o}]\!]^{g[1\to x]} = \lambda F.\ \iota z[F'(z)]$ (L)

9. $[\![\text{dono dele}_2]\!]^{g[1\to x]} = [\![\text{dono}]\!]^{g[1\to x]}([\![\text{dele}_2]\!]^{g[1\to x]})$ (AF)

10. $[\![\text{dono}]\!]^{g[1\to x]} = \lambda x.\lambda y.\ \textsc{dono}(y,x)$ (L)

11. $[\![\text{dele}_2]\!]^{g[1\to x]} = [\![\text{ele}_2]\!]^{g[1\to x]}$ (NNR, *de* é vácua)

12. $[\![\text{ele}_2]\!]^{g[1\to x]} = g[1 \to x](2) = g(2)$ (L, Ass.Mod)

13. $[\![\text{dono dele}_2]\!]^{g[1\to x]} = \lambda y.\ \textsc{dono}(y,g(2))$ (9,10,11,12)

14. $[\![\text{o dono dele}_2]\!]^{g[1\to x]} = \iota z[\textsc{dono}(z,g(2))]$ (7,8,13)

15. $[\![\text{mordeu o dono dele}_2]\!]^{g[1\to x]} = \lambda y.\ \textsc{mordeu}(y, \iota z[\textsc{dono}(z, g(2))])$ (5,6,14)

16. $[\![\text{t}_1]\!]^{g[1\to x]} = g[1 \to x](1) = x$ (L,Ass.Mod)

17. $[\![\text{S}]\!]^{g[1\to x]} = 1$ *sse* $\textsc{mordeu}(x, \iota z[\textsc{dono}(z, g(2))])$ (4,15,16)

18. $[\![\text{S}']\!]^g = \lambda x.\ \textsc{mordeu}(x, \iota z[\textsc{dono}(z, g(2))])$ (2,17)

19. $[\![\text{S}'']\!]^g = [\![\text{que}]\!]^g(\lambda x.\ \textsc{mordeu}(x, \iota z[\textsc{dono}(z, g(2))]))$ (1,18)

20. $[\![\text{que}]\!]^g = \lambda F_{\langle e,t\rangle}.\lambda G_{\langle e,t\rangle}.\lambda x_e.\ G'(x)\ \&\ F'(x)$ (L)

21. $[\![\text{S}'']\!]^g = \lambda G_{\langle e,t\rangle}.\lambda x_e.\ G'(x)\ \&\ \textsc{mordeu}(x, \iota z[\textsc{dono}(z, g(2))])$ (19,20)

Combinando o resultado acima com o a extensão do nome *cachorro*, obteremos o resultado que desejávamos para o NP *cachorro que mordeu o dono dele*:

$[\![\text{NP}]\!]^g = \lambda x_e.\ \textsc{cachorro}(x)\ \&\ \textsc{mordeu}(x, \iota x[\textsc{dono}(x, g(2))])$

Na outra leitura possível para (15), quem fugiu foi um cachorro que mordeu o próprio dono. Essa leitura pode ser capturada se coindexarmos o pronome relativo, o vestígio na posição de sujeito e o pronome *ele*:

(18) $[_{\text{NP}}$ cachorro [que$_1$ $[_{\text{S}}$ t_1 mordeu o dono dele$_1$]]]

Nesse caso, dada a coindexação entre o vestígio, o pronome *ele* e o pronome relativo, obteremos o seguinte:

$[\![\text{t}_1]\!]^{g[1\to x]} = x$

$[\![\text{dele}_1]\!]^{g[1\to x]} = x$

$[\![\text{S}]\!]^{g[1\to x]} = 1$ *sse* $\textsc{mordeu}(x, \iota z[\textsc{dono}(z, x)])$

O restante da derivação procede de maneira análoga ao que vimos acima, resultando na seguinte extensão para o NP *cachorro que mordeu o dono dele*:

$[\![\text{NP}]\!]^g = \lambda x_e.\ \textsc{cachorro}(x)\ \&\ \textsc{mordeu}(x, \iota z[\textsc{dono}(z, x)])$

Conforme desejávamos, obtivemos uma extensão que não depende da atribuição.

Sugestões de leitura

Para uma cobertura abrangente da modificação (adjetival e adverbial), ver Morzycki (2015). Sobre a semântica dos adjetivos graduáveis, como *grande* e *alto*, e das construções em que aparecem, ver Murphy (2010), capítulo 11, para uma discussão informal. Para um tratamento formal, ver Kennedy (1997) e as referências lá citadas. Sobre as diversas estratégias de relativiza-

ção do português brasileiro, ver Tarallo (1990). O tratamento sintático das orações relativas via movimento e ligação de vestígios tem origem nos trabalhos de Noam Chomsky. Para uma introdução, consultar Carnie (2013), Haegeman (1994) e Mioto et al. (2005). Para análises em outros quadros teóricos que não se valem de movimento sintático, ver Sag et al. (2003) e Jacobson (1999).

Exercícios

I. Considere o DP abaixo e derive sua extensão, passo a passo, indicando os princípios composicionais utilizados.

(1) [$_{DP}$ A escritora que João detestou o livro que ela publicou]

II. Considere a sentença (2) abaixo e as duas atribuições de índices em (i) e (ii) logo a seguir:

(2) O cachorro que João comprou mordeu o gato que Maria comprou.

(i) O cachorro que$_1$ João comprou t_1 mordeu o gato que$_1$ Maria comprou t_1.

(ii) O cachorro que João comprou t_1 mordeu o gato que$_2$ Maria comprou t_2.

Há alguma diferença semântica entre essas representações? Por quê? Qual a diferença ente o que está acontecendo nesses casos e o que vimos com o exemplo (15) e as representações em (16) e (18)?

6 Quantificação nominal

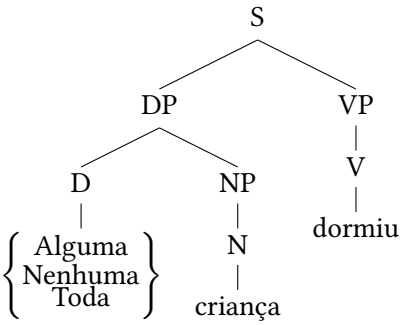
Neste capítulo, trataremos do valor semântico de expressões como *alguém*, *ninguém*, *todo mundo*, *algum cachorro*, *um livro*, *nenhuma criança* e *todo professor*. Essas expressões são chamadas de EXPRESSÕES QUANTIFICADORAS ou QUANTIFICACIONAIS. Como se pode ver pelas sentenças abaixo, a exemplo das expressões referenciais que já estudamos nos capítulos anteriores, as expressões quantificadoras também ocupam, pelo menos superficialmente, posições argumentais de predicados verbais e não verbais:

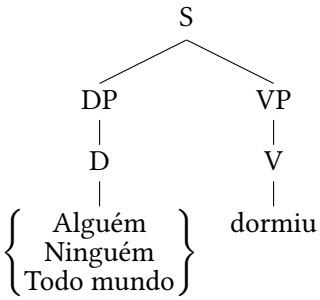
(1) a. Alguém dormiu.
    b. Ninguém está em casa.
    c. Todo Mundo ama alguém.
    d. Algum cachorro latiu.
    e. Nenhuma criança gosta do João.
    f. O João entregou um livro pra todo professor.

Entretanto, apesar dessa semelhança sintática, expressões referenciais e expressões quantificadoras devem receber tratamentos semânticos distintos, conforme veremos já na próxima seção. No curso da análise que faremos, colocaremos em uso uma série de recursos já utilizados nos capítulos anteriores, como atribuições, índices, movimento sintático e regras de mudança de tipos.

6.1 Expressões quantificadoras e indivíduos

Do ponto de vista sintático, trataremos as expressões quantificadoras como DPs. Para aquelas sintaticamente complexas, como *alguma*/*nenhuma*/*toda criança*, assumiremos que os NPs são complementos dos determinantes (Ds) *alguma*, *nenhuma* e *toda*, respectivamente. Já as expressões *todo mundo*, *alguém* e *ninguém* serão tratadas como itens lexicais simples, todas elas pertencentes à categoria D.





Até o início deste capítulo, já havíamos analisado três tipos de DPs: descrições definidas, como *o prefeito de Campinas*, nomes próprios, como *João*, e pronomes, como *ele* ou *ela*. Para todos eles, atribuímos o tipo semântico e a suas extensões, ou seja, assumimos que eram indivíduos, membros de D_e.

A semelhança sintática entre esses DPs e os DPs quantificadores ilustrados acima pode nos trazer a tentação de tratá-los semanticamente da mesma forma, ou seja, como expressões referenciais. Nossa tarefa, nesse caso, seria descobrir a que indivíduos correspondem as extensões dos DPs quantificadores.

Para o caso de *alguém* ou *algum menino*, a resposta pode parecer clara: assim como a extensão de *João* é o indivíduo João, a extensão de *alguém* é também um indivíduo, só que neste caso não sabemos, ou não revelamos, qual. Podemos chamá-lo simplesmente de *i*. O mesmo vale para *algum menino*, cuja extensão neste caso seria um menino *m* qualquer. Conforme mostrado abaixo, as condições de verdade das sentenças abaixo seriam todas representadas da mesma forma, variando apenas o indivíduo atribuído à extensão dos sujeitos:

(2) a. $[\![[_S$ João dormiu$]]\!]$ = $[\![$dormiu$]\!]([\![$João$]\!])$ = 1 *sse* João dormiu.
b. $[\![[_S$ Alguém dormiu$]]\!]$ = $[\![$dormiu$]\!]([\![$alguém$]\!])$ = 1 *sse* i dormiu.
c. $[\![[_S$ Algum menino dormiu$]]\!]$ = $[\![$dormiu$]\!]([\![$algum menino$]\!])$ = 1 *sse* m dormiu.

Essa semelhança nos leva, entretanto, a certas previsões que não são confirmadas na prática. Considere, por exemplo, a sentença (3) abaixo:

(3) Não é verdade que Maria esteja namorando João, mas ela está namorando Pedro.

(3) não é uma contradição, já que o fato de Maria não estar namorando João é plenamente compatível com o fato de ela estar namorando Pedro. Considere agora (4):

(4) Não é verdade que Maria esteja namorando alguém, mas ela está namorando Pedro.

(4) soa contraditória, e isso seria esperado se a primeira parte da sentença fosse interpretada como afirmando que não existe nenhum indivíduo x, tal que Maria esteja namorando x. Isto implica que ela não está namorando o Pedro, o que contradiz diretamente a segunda parte de (4). Entretanto, se o que a primeira parte de (4) afirma é que Maria não está namorando um certo indivíduo i, cuja identidade não é revelada, ficamos sem uma explicação de porque (4) soa contraditória, já que ela não estar namorando um certo indivíduo é plenamente compatível com ela estar namorando o Pedro, bastando que esse indivíduo não seja o próprio Pedro.

Considere, agora, as duas inferências abaixo:

(5) João cantou e João dançou. Logo, João cantou e dançou.

(6) # Alguém cantou e alguém dançou. Logo, alguém cantou e dançou.

Enquanto a inferência em (5) é claramente válida, a inferência em (6) é claramente inválida, o que indicamos por # acima. Por exemplo, se Maria foi a única pessoa que cantou, e Joana a única que dançou, então estaríamos dizendo a verdade ao dizer que alguém cantou e alguém dançou, mas estaríamos mentindo se disséssemos que alguém cantou e dançou. Mas se *alguém* denota um indivíduo, (6), assim como (5), deveria ser uma inferência válida. Por quê? Se lembrarmo-nos do que vimos sobre coordenação no capítulo 3, veremos que nosso sistema atribui tanto à premissa quanto à conclusão em (5) as mesmas condições de verdade, o que é desejável, já que essas sentenças são, de fato, equivalentes. Mas, se assumirmos que a extensão de *alguém* é também um indivíduo, seja ele qual for, então estaremos comprometidos com a equivalência da premissa e da conclusão em (6), o que não é desejável, já que, como acabamos de ver, essas sentenças não têm as mesmas condições de verdade.

Vejamos, agora, o caso de *ninguém*. Aqui, já não é nada óbvio que indivíduo poderia ser a extensão desse item. Mas ainda assim, assumamos que exista um tipo especial de indivíduo (indivíduo nulo?) que corresponda à extensão de *ninguém*. Com isso em mente, considere as inferências abaixo:

(7) a. João cantou e dançou. Logo, João cantou e João dançou.

b. # Ninguém cantou e dançou. Logo, ninguém cantou e ninguém dançou.

Novamente, não temos um paralelo entre nomes próprios e sintagmas quantificadores: (7a) é válida, mas (7b) não o é. Para verificar, basta voltar ao nosso cenário anterior. Nele não havia ninguém que cantou e dançou, mas havia alguém que cantou e também alguém que dançou. Logo, naquele cenário, a premissa em (7b) seria verdadeira, mas a conclusão falsa. Infelizmente, se atribuirmos a *ninguém* um indivíduo como extensão, nosso sistema atribuirá à premissa e à conclusão em (7b) as mesmas condições de verdade, inviabilizando uma explicação para a não validade da inferência.

Por fim, considerações semelhantes se aplicam à expressão *todo mundo*, a cuja extensão poderíamos pensar em atribuir um indivíduo correspondente ao conjunto ou à soma de todos os indivíduos comuns, como João, Pedro, Maria, etc.

(8) a. João comeu peixe ou carne. Logo, João comeu peixe ou João comeu carne.

b. # Todo mundo comeu peixe ou carne. Logo, todo mundo comeu peixe ou todo mundo comeu carne.

Mais uma vez, temos razões para não identificarmos a extensão de um sintagma quantificador com um indivíduo, seja ele qual for, justamente pelo comportamento semântico díspar entre as expressões.

6.2 Quantificadores generalizados

Abandonemos então a ideia de tratar as extensões dos sintagmas quantificadores como indivíduos e busquemos uma alternativa.

Comecemos com a sentença abaixo:

(9) Alguém dormiu.

Intuitivamente, esta sentença é verdadeira se existir um indivíduo, seja ele quem for, que tenha dormido, e falsa se não houver nenhum que tenha. Nós já sabemos que a extensão do VP *dormiu* é uma função (tipo $\langle e, t\rangle$) que leva um indivíduo x no valor de verdade 1, se x dormiu, e no valor de verdade 0, se x não dormiu. Podemos imaginar a contribuição de *alguém* para o significado da sentença como um dispositivo que faz uma busca na extensão de VP e verifica se há algum indivíduo que é levado ao valor de verdade 1. Se houver, a sentença será verdadeira. Caso contrário, ela será falsa. Vamos formalizar esta ideia, atribuindo à extensão de *alguém* o tipo $\langle\langle e, t\rangle, t\rangle$. A extensão de VP servirá como argumento para esta função, que retornará o valor de verdade 1, se, e somente se,

algum indivíduo no domínio da extensão de VP for levado ao valor 1.

$[\![\text{alguém}]\!] = \lambda F_{\langle et\rangle}.\ \exists x \in D_e\ [F'(x)]$

Uma expressão da forma $\exists x \in D_e\ [\phi]$ deve ser lida como "existe um x pertencente a D_e, tal que ϕ". O símbolo $\exists$ é tomado emprestado da lógica de predicados e chamado de QUANTIFICADOR EXISTENCIAL. Frequentemente, vamos omitir a informação sobre o domínio, já implícita no uso de variáveis como x, y e z, na metalinguagem, para membros de D_e, além de F, G e H para funções de tipo $\langle e, t\rangle$. Para obter as condições de verdade de nossa sentença, basta então que utilizemos aplicação funcional:

$[\![[_{\text{S}}\ \text{Alguém dormiu}]]\!] = [\![[_{\text{DP}}\ \text{Alguém}]]\!]([\![[_{\text{VP}}\ \text{dormiu}]]\!])$

$= (\lambda F.\ \exists x[F'(x)])(\lambda x.\ \textsc{dormiu}(x))$

$= 1$ *sse* $\exists x[\textsc{dormiu}(x)]$

Note que semanticamente foi o VP que serviu de argumento ao DP sujeito. Podemos estender o tratamento que acabamos de dar para *alguém* a *ninguém*. Considere a sentença abaixo:

(10) Ninguém dormiu

A contribuição de *ninguém* pode ser vista como um dispositivo que faz uma busca na extensão de VP e verifica se há algum indivíduo que é levado ao valor de verdade 1. Se não houver, a sentença será verdadeira. Caso contrário, ela será falsa:

$[\![\text{ninguém}]\!] = \lambda F.\ \neg\exists x[F'(x)]$

A expressão $\neg\exists x$ deve ser lida como "é falso que exista um x". A derivação das condições de verdade de (10) procede da seguinte forma:

$[\![[_{\text{S}}\text{Ninguém dormiu}]]\!] = [\![[_{\text{DP}}\text{Ninguém}]]\!]([\![[_{\text{VP}}\text{dormiu}]]\!])$

$= (\lambda F'.\ \neg\exists x[F'(x)])(\lambda x.\ \textsc{dormiu}(x))$

= 1 *sse* $\neg\exists x[\textsc{dormiu}(x)]$

Note as condições de verdade opostas de *Ninguém dormiu* e *Alguém dormiu*. Isso reflete a nossa intuição de que se uma dessas sentenças for verdadeira, a outra será falsa e vice-versa.

O quantificador *todo mundo* pode ser tratado nessa mesma linha. Para que a sentença *Todo mundo dormiu* seja verdadeira, é preciso que todos os membros de D_e sejam levados ao valor de verdade 1 pela extensão do VP *dormiu*, ou seja, para todo indivíduo *x*, é preciso que *x* tenha dormido. A entrada lexical abaixo capta esta intuição:

$[\![\text{todo mundo}]\!] = \lambda F.\ \forall x[F'(x)]$

A expressão $\underline{\forall x}$, também oriunda da lógica de predicados, deve ser lida como "para todo *x*", sendo $\forall$ conhecido como QUANTIFICADOR UNIVERSAL. A derivação a seguir mostra como as condições de verdade de *Todo mundo dormiu* são obtidas através de aplicação funcional:

$[\![[_{\text{S}}\text{Todo mundo dormiu}]]\!] = [\![[_{\text{DP}}\text{Todo mundo}]]\!]([\![[_{\text{VP}}\text{dormiu}]]\!])$

$= (\lambda F.\ \forall x[F'(x)])(\lambda x.\ \textsc{dormiu}(x))$

= 1 *sse* $\forall x[\textsc{dormiu}(x)]$

Oferecemos, assim, um tratamento uniforme aos itens *ninguém*, *alguém* e *todo mundo*. Todos foram tratados como funções do tipo $\langle\langle e, t\rangle, t\rangle$. Funções deste tipo são conhecidas na literatura como QUANTIFICADORES GENERALIZADOS.

6.3 Determinantes quantificadores

Comparemos o significado dos pares de sentenças abaixo:

(11) a. Ninguém dormiu.
b. Nenhum menino dormiu.

(12) a. Alguém dormiu.
b. Algum menino dormiu.

(13) a. Todo mundo dormiu.

b. Todo menino dormiu.

Vamos começar com as sentenças em (11). Para que (11a) seja verdadeira, é necessário que não haja nenhum indivíduo que tenha dormido, caso contrário a sentença será falsa. Já (11b) é menos exigente. Para que ela seja verdadeira, basta que nenhum menino tenha dormido, ou seja, não deve haver nenhum indivíduo que seja ao mesmo tempo um menino e que tenha dormido. O fato de haver ou não meninas ou adultos que tenham dormido é irrelevante neste caso. As condições de verdade de (11b) podem ser representadas da seguinte forma:

$[\![(11b)]\!] = 1$ *sse* $\neg\exists x[\textsc{menino}(x) \,\&\, \textsc{dormiu}(x)]$

Para que obtenhamos esse resultado composicionalmente a partir das extensões do DP *nenhum menino* e do VP *dormiu*, basta que modifiquemos minimamente a extensão proposta mais acima para *ninguém*:

$[\![\text{nenhum menino}]\!] = \lambda F.\ \neg\exists x[\textsc{menino}(x) \,\&\, F'(x)]$

Assim como no caso de *ninguém*, a extensão de *nenhum menino* é também um quantificador generalizado, ou seja, uma função de tipo $\langle\langle e, t\rangle, t\rangle$. Para VP, não há nada de novo a ser dito. Sua extensão é de tipo $\langle e, t\rangle$:

$[\![[_{\text{VP}}\ \text{dormiu}]]\!] = \lambda x.\ \textsc{dormiu}(x)$

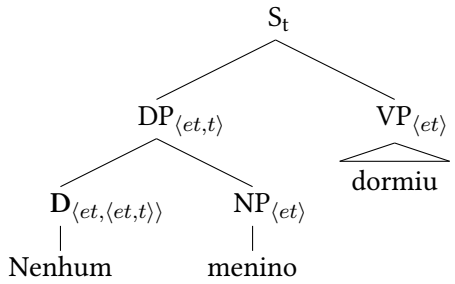
Utilizando aplicação funcional, chegamos ao resultado desejado:

$[\![[_{\text{S}}\ \text{Nenhum menino dormiu}]]\!] = [\![[_{\text{DP}}\ \text{Nenhum menino}]]\!]([\![[_{\text{VP}}\ \text{dormiu}]]\!])$

$= (\lambda F.\ \neg\exists x[x\ \textsc{menino}(x) \,\&\, F'(x)])(\lambda x.\ \textsc{dormiu}(x))$

$= 1$ *sse* $\neg\exists x[\textsc{menino}(x) \,\&\, \textsc{dormiu}(x)]$

Vejamos agora como obter a extensão de *nenhum menino* a partir de seus constituintes imediatos: o NP *menino* e o determinante *nenhum*. Já sabemos que a extensão do NP é de tipo $\langle e, t\rangle$. Acabamos de propor que a extensão do DP é de tipo $\langle\langle e, t\rangle, t\rangle$. Se quisermos utilizar aplicação funcional para obter a extensão de DP, só nos resta uma alternativa para o tipo semântico de D: $\langle\langle e, t\rangle, \langle\langle e, t\rangle, t\rangle\rangle$, ou, de maneira simplificada, $\langle et, \langle et, t\rangle\rangle$:



Em termos informais, o que a extensão de *nenhum* faz é o seguinte: ela olha para a extensão de NP e, em seguida, para a extensão de VP, verificando se existe algum indivíduo que é levado ao valor de verdade 1 tanto pela extensão de NP quanto pela extensão de VP. Se não houver, a sentença será verdadeira. Se houver, ela será falsa. Em termos formais, temos o seguinte:

$[\![\text{nenhum}]\!] = \lambda F.\ \lambda G.\ \neg\exists x[F'(x)\ \&\ G'(x)]$

Observando o tipo semântico dessa extensão do determinante ($\langle et, \langle et, t\rangle\rangle$), nota-se uma semelhança com o tipo das extensões dos verbos transitivos, como *amar* ($\langle e, \langle e, t\rangle\rangle$). Enquanto a extensão de um verbo transitivo mapeia dois indivíduos (tipo e) em um valor de verdade, a de um determinante mapeia duas funções (tipo $\langle e, t\rangle$) em um valor de verdade. Assim como as extensões dos verbos transitivos podem ser vistas como relações entre indivíduos, as dos determinantes podem ser vistas como relações entre funções. Essas relações costumam ser chamadas de RELAÇÕES DE SEGUNDA ORDEM, distinguindo-se das RELAÇÕES DE PRIMEIRA ORDEM associadas aos verbos transitivos.

Prosseguindo na interpretação composicional dos DPs, podemos utilizar aplicação funcional:

$[\![[_{\text{DP}}\ \text{Nenhum menino}]]\!] = [\![[_{\text{D}}\ \text{Nenhum}]]\!]([\![[_{\text{NP}}\ \text{menino}]]\!])$

$= (\lambda F.\ \lambda G.\ \neg\exists x[F'(x)\ \&\ G'(x)])([\![\text{NP}]\!])$

$= \lambda G.\ \neg\exists x[\textsc{menino}(x)\ \&\ G'(x)]$

Para postularmos extensões para os determinantes *algum* e *todo*, procedemos de maneira análoga à que acabamos de ver para o caso de *ninguém*. Comparando o significado de (12a) com o de (12b), podemos observar que, enquanto a primeira é verdadeira sempre que houver algum indivíduo que esteja acordado, a segunda

é mais exigente, precisando haver algum menino acordado. Intuitivamente, o que a extensão de *algum* parece fazer é olhar para as extensões de NP e de VP e verificar se existe um indivíduo que é levado ao valor de verdade 1 tanto pela extensão de NP quanto pela extensão de VP. Se houver, a sentença será verdadeira, caso contrário será falsa.

$[\![\text{algum}]\!] = \lambda F.\ \lambda G.\ \exists x[F'(x)\ \&\ G'(x)]$

Para o caso do determinante *todo*, se compararmos o significado de (13a) com o de (13b), notaremos que as condições de verdade da segunda são menos exigentes que as da primeira. Em termos informais, o que a extensão de *todo* faz é olhar para as extensões de NP e de VP e verificar se todo indivíduo levado ao valor de verdade 1 pela extensão de NP é também levado ao valor de verdade 1 pela extensão de VP. Somente em caso afirmativo a sentença será verdadeira:

$[\![\text{todo}]\!] = \lambda F.\ \lambda G.\ \forall x[F'(x) \rightarrow G'(x)]$

A expressão $\forall x[F'(x) \rightarrow G'(x)]$ deve ser lida como "para todo x, se $F(x)$ for igual a 1, então $G(x)$ será igual a 1." A derivação das condições de verdade de *Todo menino dormiu* está esquematizada abaixo:

$[\![[_{\text{DP}}\ \text{Todo menino}]]\!] = [\![[_{\text{D}}\ \text{Todo}]]\!]([\![[_{\text{NP}}\ \text{menino}]]\!])$

$= (\lambda F.\ \lambda G.\ \forall x[F'(x) \rightarrow G'(x)])([\![\text{NP}]\!])$

$= \lambda G.\ \forall x[\textsc{menino}(x) \rightarrow G'(x)]$

$[\![[_{\text{S}}\ \text{Todo menino dormiu}]]\!] = [\![[_{\text{DP}}\ \text{Todo menino}]]\!]([\![[_{\text{VP}}\ \text{dormiu}]]\!])$

$= (\lambda F.\ \forall x[\textsc{menino}(x) \rightarrow F'(x)])(\lambda x.\ \textsc{dormiu}(x))$

$= 1$ *sse* $\forall x[\textsc{menino}(x) \rightarrow \textsc{dormiu}(x)]$

Antes de finalizar essa seção, um breve interlúdio técnico sobre a notação que empregamos acima na representação da extensão de *todo*. Essa representação fez uso do símbolo $\forall$, o quantificador universal, que já havíamos visto, e do símbolo $\rightarrow$, conhecido na lógica como IMPLICAÇÃO MATERIAL ou simplesmente IMPLICAÇÃO. Esse símbolo conecta duas fórmulas ϕ e ψ, sendo $\phi \rightarrow \psi$ lida como "ϕ im-

plica ψ", ou "se ϕ, então ψ". Assim como o & que utilizamos para a coordenação aditiva de sentenças, $\rightarrow$ é definido na lógica como um conectivo verifuncional, com o valor de verdade de $\phi \rightarrow \psi$ dependendo apenas dos valores de verdade de ϕ e ψ. Mais precisamente, sua contribuição costuma ser representada através da tabela abaixo:

(14) Tabela de verdade de $\rightarrow$

ϕ	ψ	$\phi \rightarrow \psi$
1	1	1
1	0	0
0	1	1
0	0	1

Pela tabela acima, nota-se que $\phi \rightarrow \psi$ só será falsa no caso de ϕ (chamado de ANTECEDENTE) ser verdadeiro e ψ (chamado de CONSEQUENTE) ser falso. Daí a paráfrase, se ϕ, então ψ. A ideia é que a verdade do antecedente leve à verdade do consequente. O que pode parecer estranho na tabela acima são as duas últimas linhas. Elas nos informam que se o antecedente for falso, a implicação será sempre verdadeira, independentemente da verdade ou falsidade do consequente. Para o nosso propósito, que é a representação da extensão do determinante quantificador *todo* vista mais acima, isso significa que em um cenário em que não houvesse meninos, por exemplo, a sentença *todo menino dormiu* seria verdadeira. De fato, essa não parece uma consequência desejável. Tampouco seria o seu oposto, ou seja, assumir que a implicação seja falsa nesses casos. Talvez caiba aqui uma análise pressuposicional em que sentenças da forma [todo NP VP] só tenham valor de verdade quando a extensão de NP levar ao menos um indivíduo ao valor 1. Ou, talvez, essas sentenças requeiram uma combinação de recursos semânticos e pragmáticos de modo que seu uso apropriado imponha condições sobre a extensão de NP. No restante deste capítulo, manteremos a entrada lexical vista mais acima para *todo*, deixando de lado essas (importantes) complicações (mas ver as sugestões de leitura ao final do capítulo).

6.4 Nomes próprios e descrições definidas como quantificadores generalizados

Já vimos mais acima que não parece uma boa ideia tratar as extensões de DPs quantificadores como indivíduos. Vamos aqui percorrer o caminho oposto e tentar a unificação dos tipos semânticos dos DPs, assumindo que nomes próprios

e descrições definidas, a exemplo dos DPs quantificadores, são também de tipo $\langle\langle e, t\rangle, t\rangle$. Em outras palavras, vamos investigar a possibilidade de tratarmos nomes próprios e descrições definidas como quantificadores generalizados. Posteriormente, tiraremos vantagem dessa unificação.

Considere os exemplos abaixo:

(15) João dormiu.

(16) O prefeito de Campinas dormiu.

A ideia é ver a contribuição do nome próprio João para o significado de (15) como um dispositivo que olha para a extensão do VP *dormiu* e verifica se o indivíduo João é levado ou não ao valor de verdade 1. Se for, então a sentença será verdadeira. Se não for, a sentença será falsa. Formalizaremos esta ideia, assumindo a seguinte extensão para *João*:

$[\![\text{João}]\!] = \lambda F_{\langle e,t\rangle}.\ F'(\textit{joão})$

As condições de verdade de (15) são obtidas através de aplicação funcional. Neste caso, a extensão de VP é que serve de argumento para a extensão do nome próprio:

$[\![[_{\text{S}}\text{João dormiu}]]\!] = [\![[_{\text{DP}}\text{João}]]\!]([\![[_{\text{VP}}\text{dormiu}]]\!])$

$= (\lambda F.\ F'(\textit{joão}))(\lambda x.\ \textsc{dormiu}(\textit{joão}))$

$= 1$ *sse* $\textsc{dormiu}(\textit{joão})$

Essas são, claro, as mesmas condições que já havíamos obtido anteriormente, tratando as extensões dos nomes próprios como indivíduos (tipo *e*).

A sentença (16) pode ser tratada de maneira semelhante. A extensão de *o prefeito de Campinas* toma a extensão do VP *dormiu* como argumento e retorna o valor de verdade 1 se, e somente se, o único indivíduo que é prefeito de Campinas tiver dormido.

$[\![\text{[o prefeito de Campinas]}]\!] = \lambda F.\ F'(\iota x[\textsc{prefeito}(x,\textit{Campinas})])$

As condições de verdade de (16) são obtidas de maneira idêntica à que acabamos de ver para (15). Já a extensão do artigo definido precisa ser modificada de

acordo, uma vez que, agora, esse será tratado como os determinantes quantificadores *algum*, *nenhum* e *todo*. Seu novo tipo semântico será $\langle et, \langle et, t\rangle\rangle$:

$[\![\text{o}]\!] = \lambda F : \exists!x[F'(x)].\ \lambda G.\ G'(\iota y[F'(y)])$

Note que esta entrada lexical preserva a análise pressuposicional que havíamos adotado anteriormente para o artigo definido. Para obter as condições de verdade de (16), utilizamos aplicação funcional duas vezes. Primeiro para obter a extensão de DP, a partir de D e NP. Depois para obter a extensão de S, a partir de DP e VP:

$[\![[_{\text{DP}}\text{o prefeito de Campinas}]]\!] = [\![[_{\text{D}}\ \text{o}]]\!]([\![[_{\text{NP}}\ \text{prefeito de Campinas}]]\!])$

$[\![(16)]\!] = [\![[_{\text{DP}}\ \text{o prefeito de Campinas}]]\!]([\![[_{\text{VP}}\ \text{dormiu}]]\!])$

Os detalhes da derivação ficam para o leitor como exercício, que poderá observar a adequação das condições de verdade obtidas.

A conclusão a que chegamos é que é possível tratar uniformemente a extensão de nomes próprios, descrições definidas e DPs quantificadores como sendo quantificadores generalizados. Como já anunciamos, veremos mais adiante uma vantagem que essa unificação traz. Antes, porém, vamos aumentar nosso estoque de quantificadores generalizados.

6.5 Outros DPs quantificadores

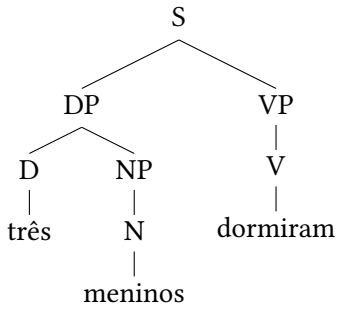
Vejamos, agora, outros DPs cujas extensões podem ser analisadas como quantificadores generalizados, bem como os determinantes que são seus núcleos. Considere as sentenças abaixo:

(17) Três meninos dormiram.

(18) Mais de três meninos dormiram.

(19) Menos de três meninos dormiram.

Comecemos por (17), para a qual atribuiremos a seguinte estrutura:



Vamos assumir que o plural na forma verbal *dormiram* resulta de uma regra sintática de concordância sujeito-verbo que não afeta a interpretação do verbo e que, portanto, a extensão de *dormiram* é a mesma de *dormiu*. Vamos assumir, ainda, que o plural no nome *meninos* também resulta de uma regra sintática de concordância, desta vez entre um determinante D e o NP que é seu complemento. Portanto, estamos assumindo que a extensão de *meninos* é igual à extensão de *menino*.

Figurativamente, podemos pensar na extensão do DP *três meninos* como um contador que verifica o número de meninos que são levados pela extensão de VP ao valor de verdade 1. Se esse número for três, a sentença será verdadeira. Se o número for diferente de três, a sentença será falsa. Para formalizar essa ideia, vamos lançar mão do conceito de cardinalidade da teoria dos conjuntos. Como estaremos lidando apenas com conjuntos finitos, definiremos a cardinalidade de um conjunto como o número de elementos pertencentes ao conjunto. Usaremos a notação |C| para representar a cardinalidade de um conjunto C qualquer. Com isso em mente, considere a extensão abaixo:

$$[\![\text{três meninos}]\!] = \lambda F.\ |\ \{x|\textsc{menino}(x)\} \cap \{x|F'(x)\}\ |\ =\ 3$$

Essa extensão é um quantificador generalizado, tipo $\langle et, t\rangle$, que, ao tomar uma função F como argumento, retorna o valor de verdade 1 se, e somente se, a intersecção do conjunto dos meninos com o conjunto dos indivíduos levados por F ao valor 1 tiver três elementos. Lembre-se de que esse último conjunto é o conjunto caracterizado pela função F e que, conforme decidimos no capítulo 2, representamos por F_C:

$$[\![\text{três meninos}]\!] = \lambda F.\ |\ \{x|\textsc{menino}(x)\} \cap F_C\ |\ =\ 3$$

Vamos verificar a adequação dessa extensão calculando as condições de verdade

de (17). Como a extensão de VP é de tipo $\langle e, t\rangle$, podemos nos valer de aplicação funcional:

$[\![\text{S}]\!] = [\![\text{DP}]\!]([\![\text{VP}]\!])$

$[\![\text{S}]\!] = (\lambda F.\ |\ \{x|\textsc{menino}(x)\} \cap F_C\ |\ =\ 3)(\lambda x.\ \textsc{dormiu}(x))$

$[\![\text{S}]\!] = 1$ *sse* $|\ \{x|\textsc{menino}(x)\} \cap \{x|\textsc{dormiu}(x)\}\ |\ =\ 3$

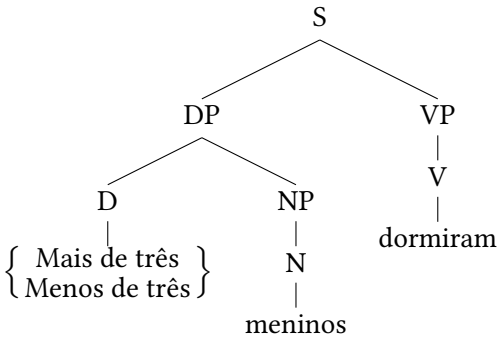
As condições acima nos dizem que (17) é verdadeira se, e somente se, a interseção do conjunto dos meninos com o conjunto dos indivíduos que dormiram tiver três elementos. Isso é exatamente o que queríamos.

Para completar nossa análise da interpretação de (17), precisamos definir a extensão do determinante *três*. Dado o que já vimos nas seções anteriores sobre o tipo semântico de outros determinantes quantificadores e o que acabamos de ver sobre a extensão do DP *três meninos*, nossa tarefa se torna simples:

$[\![\text{três}]\!] = \lambda F.\lambda G.\ |\ F_C \cap G_C\ |\ =\ 3$

Em palavras, o determinante *três*, após tomar duas funções F e G como argumentos, retornará o valor 1 se, e somente se, o número de elementos da intersecção do conjunto caracterizado por F com o conjunto caracterizado por G for igual a três. As duas funções em questão são as extensões do NP que é complemento de D e do VP do qual o DP resultante é sujeito.

Esse tratamento se estende imediatamente aos exemplos (18) e (19) acima. Do ponto de vista sintático, vamos simplificar e assumir que as expressões *mais de três* e *menos de três* correspondem a itens lexicais (obviamente, isso é apenas uma simplificação, mas foge ao escopo deste livro uma análise morfossintática mais refinada dessas expressões):



$[\![\text{mais de três}]\!] = \lambda F.\lambda G.\ |\ F_C \cap G_C\ | > 3$

$[\![\text{menos de três}]\!] = \lambda F.\lambda G.\ |\ F_C \cap G_C\ | < 3$

O leitor deve ter em mente que essa nossa breve incursão na semântica dos numerais (modificados ou não) apenas arranha a superfície de um terreno vasto e complexo. Há usos predicativos e argumentais desses numerais que certamente não se encaixam no que acabamos de ver. Sentenças como *Os três meninos chegaram* ou *três é um número primo* exigem um tratamento diferente. Além disso, mesmo os casos que apresentamos acima têm sido objeto de controvérsias semânticas que não estamos abordando aqui (ver as sugestões de leitura).

A análise dos numerais como determinantes cardinais pode ser extendida para o indefinido plural *alguns*. Para que uma sentença como (20) seja verdadeira, é preciso que mais de um menino tenha dormido, ou seja, a interseção do conjunto dos meninos com o conjunto dos indivíduos que dormiram deve ter mais de um elemento.

(20) Alguns meninos dormiram.

$[\![\text{alguns meninos}]\!] = \lambda F.\ |\ \{x|\textsc{menino}(x)\} \cap F_C\ | > 1$

$[\![\text{alguns}]\!] = \lambda F.\lambda G\ |\ F_C \cap G_C\ | > 1$

Outros DPs cujas extensões podem receber um tratamento adequado como quantificadores generalizados são expressões da forma *a maioria dos NP*, como na sentença abaixo:

(21) A maioria dos meninos dormiu.

Aqui vamos nos esquivar da análise interna destas expressões, limitando nossa atenção ao DP como um todo. Para que (21) seja verdadeira, é preciso que o número de meninos que dormiram seja superior ao número de meninos que não dormiram. Desse modo, garantimos que mais da metade dos meninos dormiu, que é o que desejamos. Novamente, serviremo-nos do conceito de cardinalidade:

$[\![\text{a maioria dos meninos}]\!] = \lambda F.\ |\ [\![\text{meninos}]\!]_C \cap F_C\ | > |\ [\![\text{meninos}]\!]_C \cap \overline{F_C}\ |$

Lembre-se que $\overline{F_C}$ é o complemento de F_C, o conjunto caracterizado por F. De maneira mais geral, temos que:

$[\![\text{a maioria dos NP}]\!] = \lambda F.\ |\ [\![\text{NP}]\!]_C \cap F_C\ | > |\ [\![\text{NP}]\!]_C \cap \overline{F_C}\ |$

E se, simplificadamente, tratarmos *a maioria dos* como um determinante que toma um NP por complemento, teríamos:

$[\![\text{a maioria dos}]\!] = \lambda F.\ \lambda G.\ |\ F_C \cap G_C\ | > |\ F_C \cap \overline{G_C}\ |$

Fica a cargo do leitor verificar a adequação desta extensão, calculando as condições de verdade de (21).

6.6 Coordenação de DPs

Discutimos mais acima como tratar as extensões de nomes próprios e descrições definidas como quantificadores generalizados. Isto permitiu uma unificação dos DPs no que diz respeito ao tipo semântico de suas extensões: todas elas são de tipo $\langle et, t\rangle$. Uma vantagem dessa unificação é que ela torna possível uma análise para a coordenação de DPs baseada no que já estudamos no capítulo sobre coordenação e, sobretudo, na entrada lexical que propusemos para o conectivo *e*. Para entender como, considere as seguintes sentenças:

(22) João e Maria tiraram 10 na prova.

(23) Maria e alguns meninos tiraram 10 na prova.

(24) Dois meninos e três meninas tiraram 10 na prova.

(25) O filho do João e algumas meninas tiraram 10 na prova.

(26) Maria e o filho do João tiraram 10 na prova.

Como mostram os exemplos acima, é possível coordenar dois tipos quaisquer de DPs, sejam eles nomes próprios, descrições definidas ou DPs quantificadores como os que vimos nas seções acima. O resultado é um outro DP, que, nos exemplos acima, funciona como sujeito do VP *tiraram 10 na prova*. Antes de passarmos à análise desses DPs, note que todas as sentenças acima podem ser parafraseadas por sentenças envolvendo coordenação sentencial:

(27) João tirou 10 na prova e Maria tirou 10 na prova.

(28) Maria tirou 10 na prova e alguns meninos tiraram 10 na prova.

(29) Dois meninos tiraram 10 na prova e três meninas tiraram 10 na prova.

(30) O filho do João tirou 10 na prova e algumas meninas tiraram 10 na prova.

(31) Maria tirou 10 na prova e o filho do João tirou 10 na prova.

Lembre-se, agora, que, no capítulo sobre coordenação e negação, após notarmos a flexibilidade do conectivo *e* em coordenar diversos tipos de constituintes, postulamos a seguinte entrada lexical:

Versão Polimórfica da Conjunção *e*:

Caso básico: se $a = t$,

$$[\![e_{\langle a,\langle a,a\rangle\rangle}]\!] = \lambda p_t.\ \lambda q_t.\ p = 1 \ \&\ q = 1$$

Casos Derivados: se $a = \langle a_1, \langle a_2, ..., \langle a_n, t\rangle ...\rangle\rangle$ $(n \geq 1)$,

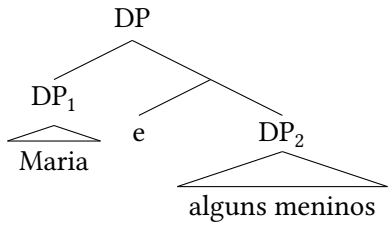
$$[\![e_{\langle a,\langle a,a\rangle\rangle}]\!] = \lambda f_a.\lambda g_a.\lambda \gamma^1_{a_1}.\lambda \gamma^2_{a_2}...\lambda \gamma^n_{a_n}.\ [\![e_{\langle t\langle t,t\rangle\rangle}]\!](f_a(\gamma^1_{a_1})(\gamma^2_{a_2})...(\gamma^n_{a_n}))(g_a(\gamma^1_{a_1})(\gamma^2_{a_2})...(\gamma^n_{a_n}))$$

Lembre-se de que tipos booleanos são tipos semânticos que terminam em *t*, como, por exemplo, t, $\langle et\rangle$, $\langle e, et\rangle$ e $\langle et, t\rangle$. Esse último é justamente o tipo dos quantificadores generalizados. Note, ainda, que a entrada lexical acima requer que o tipo semântico dos elementos coordenados seja idêntico. Isso também não é problema para nós, já que estamos justamente argumentando a favor de uma análise que atribui às extensões de todos os DPs o mesmo tipo semântico. Vejamos então o que a entrada lexical de *e* nos fornece para a coordenação envolvendo o tipo semântico $\langle et, t\rangle$.

Assumindo, no esquema acima, que $a = \langle et, t\rangle$ e efetuando as devidas substituições, obteremos o seguinte (volte ao capítulo 3 se estiver em dúvida sobre como proceder):

$[\![e_{\langle ett,\langle ett,ett\rangle\rangle}]\!] = \lambda P_{\langle et,t\rangle}.\lambda Q_{\langle et,t\rangle}.\lambda F_{\langle et\rangle}.\ P(F) = 1\ \&\ Q(F) = 1$

Vamos ilustrar o uso dessa extensão calculando as condições de verdade de (23), cujo DP sujeito *Maria e alguns meninos* tem a estrutura (simplificada) abaixo (para os demais exemplos, basta proceder de maneira análoga):



$[\![[\text{e DP}_2]]\!] = [\![\text{e}]\!]([\![\text{DP}_2]\!])$
$[\![[\text{e DP}_2]]\!] = (\lambda P.\lambda Q.\lambda F.\ P(F) = 1\ \&\ Q(F) = 1)([\![\text{DP}_2]\!])$
$[\![[\text{e DP}_2]]\!] = \lambda Q_{\langle et,t\rangle}.\lambda F_{et}.\ [\![\text{DP}_2]\!](F) = 1\ \&\ Q(F) = 1$

$[\![\text{DP}]\!] = [\![[\text{e DP}_2]]\!]([\![\text{DP}_1]\!])$
$[\![\text{DP}]\!] = \lambda F_{et}.\ [\![\text{DP}_2]\!](F) = 1\ \&\ [\![\text{DP}_1]\!](F) = 1$

Como sabemos que

$[\![\text{DP}_1]\!] = \lambda F_{\langle e,t\rangle}.\ F'(maria)$

$[\![\text{DP}_2]\!] = \lambda F_{\langle et\rangle}.\ \mid \{x \mid \textsc{menino}(x)\} \cap F_C \mid > 1$

Temos que:

$[\![\text{DP}]\!] = \lambda F_{et}.\ \mid \{x \mid \textsc{menino}(x)\} \cap F_C \mid > 1\ \&\ F'(maria)$

Para obtermos as condições de verdade de nossa sentença, basta utilizarmos aplicação funcional:

$[\![\text{S}]\!] = [\![\text{DP}]\!]([\![\text{VP}]\!])$

$[\![\text{VP}]\!] = \lambda x.\ \textsc{tirou_10_na_prova}(x)$

$[\![\text{S}]\!] = 1$ *sse* $| \{x|\textsc{menino}(x)\} \cap \{x|\textsc{tirou_10_na_prova}(x)\} | > 1$ & $\textsc{tirou_10_na_prova}(\textit{maria})$

Essas condições de verdade exigem que Maria tenha tirado 10 na prova e que mais de um menino tenha tirado 10 na prova. Isso é exatamente o que queríamos.

Antes de fecharmos esta seção, note o seguinte: ao tratarmos nomes próprios e descrições definidas como tendo extensões de tipo $\langle et, t\rangle$ não precisamos necessariamente abandonar nossas considerações dos capítulos anteriores em que elas eram de tipo *e*. Ao contrário, temos razões para manter essas extensões para que continuemos a dar conta de sentenças como *João ama Maria*, em que a extensão do verbo transitivo é de tipo $\langle e, et\rangle$, buscando, portanto, dois argumentos de tipo *e*. Para manter todos esses resultados, podemos recorrer a um expediente que já utilizamos anteriormente, a saber, uma regra de mudança de tipos. Nesse caso, nosso alvo são expressões referenciais, que queremos analisar como tendo duas possíveis extensões: uma de tipo *e* e outra de tipo $\langle et, t\rangle$. Vamos, então, manter as extensões de tipo *e* como básicas e fornecer uma regra que relacione sistematicamente essas extensões com as extensões de tipo $\langle et, t\rangle$:

Regra de Mudança de Tipos para Expressões Referenciais de tipo *e*
Seja X um DP cuja extensão α é de tipo *e*. Mude a extensão de X para α', sendo α' de tipo $\langle et, t\rangle$ e definida da seguinte forma:

$\alpha' = \lambda F_{\langle et\rangle}.\ F'(\alpha)$

Para o nome próprio *Maria*, por exemplo, a regra acima nos fornece o seguinte:

$[\![\text{Maria}_{\langle et,t\rangle}]\!] = \lambda F_{\langle et\rangle}.\ F'([\![\text{Maria}_e]\!])$
$[\![\text{Maria}_e]\!] = maria$
$[\![\text{Maria}_{\langle et,t\rangle}]\!] = \lambda F_{\langle et\rangle}.\ F'(maria)$

Preservamos, assim, todos os resultados que obtivemos até aqui no que diz respeito a nomes próprios e descrições definidas.

Cumpre, por fim, salientar que nem todos os casos de coordenação de DPs são amenos a um tratamento na linha do que vimos nesta seção. Certos predicados admitem (ou até exigem) interpretações ditas plurais ou coletivas que nosso sistema não é capaz de captar. Por exemplo:

(32) Alguns alunos e algumas alunas formaram uma equipe para a olimpíada de matemática da escola.

(33) O professor e alguns alunos se reuniram no pátio da escola.

Nesses casos, as coordenações dos DPs parecem se comportar de maneira diferente, não sendo parafraseáveis por coordenações sentenciais:

(34) Alguns alunos formaram uma equipe para a Olimpíada de Matemática da escola e algumas alunas formaram uma equipe para a Olimpíada de Matemática da escola.

(35) O professor se reuniu no pátio da escola e alguns alunos se reuniram no pátio da escola.

Para entender o que se passa em casos como esses, precisaríamos adentrar a semântica da pluralidade, algo que não faremos neste livro, limitando-nos a sugerir algumas leituras ao final do capítulo.

6.7 Conservatividade: um universal semântico

Considere as duas sentenças abaixo e reflita sobre suas condições de verdade:

(36) a. Alguns alunos da minha turma são funcionários da universidade.
b. Alguns funcionários da universidade são alunos da minha turma.

Como podemos notar, essas sentenças têm condições de verdade idênticas, ou seja, não há como uma delas ser verdadeira e a outra falsa. Isto continua sendo válido se trocarmos os NPs e VPs acima por outros, desde que mantenhamos a estrutura do par, ou seja, desde que o NP da primeira sentença corresponda ao VP da segunda e o NP da segunda corresponda ao VP da primeira:

(37) a. Alguns brasileiros são descendentes de russos.
b. Alguns descendentes de russos são brasileiros.

Essas equivalências nos dizem algo sobre a extensão do determinante *alguns*. Imagine duas funções *F* e *G* quaisquer, e que essas funções sirvam, nessa ordem, de argumento a *alguns*. As condições de verdade que obteremos estão representadas abaixo:

$[\![\text{alguns}]\!](F)(G) = 1$ *sse* $\mid F_C \cap G_C \mid > 1$

Compare agora essas condições de verdade com as condições de verdade que obteríamos se invertêssemos a ordem dos argumentos do determinante, ou seja, se a extensão de *alguns* se aplicasse primeiro a *G* e depois a *F*, sendo *F* e *G* as mesmas funções imaginadas acima:

$[\![\text{alguns}]\!](G)(F) = 1$ *sse* $\mid G_C \cap F_C \mid > 1$

Não é difícil notar que as duas condições de verdade acima são idênticas. Isso porque a operação de interseção é comutativa. Assim, para quaisquer conjuntos *A* e *B*, $A \cap B = B \cap A$. Isto torna a ordem dos argumentos do determinante *alguns* irrelevante do ponto de vista das condições de verdade. Sempre que isto for verdade para um determinante *D* qualquer, diremos que *D* é simétrico. Ou seja, estamos atribuindo uma propriedade formal, **simetria**, a um determinante. Essa propriedade pode ser definida da seguinte forma:

Simetria
Para qualquer determinante *D*, e quaisquer funções *F* e *G* de tipo $\langle e, t\rangle$, *D* é simétrico se, e somente se $[\![\text{D}]\!](\text{F})(\text{G}) = [\![\text{D}]\!](\text{G})(\text{F})$

Outros determinantes simétricos são *nenhum*, todos os numerais e também os numerais modificados como *mais de três*, *menos de quatro*, etc. No entanto, nem todos os determinantes são simétricos. Por exemplo, *todo* não o é, conforme atesta o par de sentenças a seguir:

(38) a. Todo brasileiro é sul-americano.
b. Todo sul-americano é brasileiro.

Enquanto (38a) é verdadeira, (38b) é, obviamente, falsa.

Desde as primeiras aplicações da teoria dos quantificadores generalizados voltadas para o estudo das línguas naturais, os estudiosos da área se perguntam se há propriedades que caracterizam todos os determinantes de todas as línguas

naturais. Nesta seção, vamos olhar para uma das propriedades que se tem postulado como um universal semântico. Trata-se da propriedade conhecida como CONSERVATIVIDADE.

Para se ter uma ideia mais clara sobre o que é um determinante *conservativo*, é conveniente pensar nos argumentos da extensão de um determinante como sendo conjuntos de indivíduos, em vez de funções que levam indivíduos em valores de verdade. Esta mudança, claro, é uma mera questão de ponto de vista, já que, como vimos ao final do capítulo 2, existe uma correspondência biunívoca entre conjuntos de indivíduos e funções de tipo $\langle e, t\rangle$. Considere, então, as três sentenças abaixo:

(39) Alguma criança chorou.

(40) Nenhuma criança chorou.

(41) Toda criança chorou.

Todas essas sentenças são da forma D-NP-VP. Vamos representar por N o conjunto caracterizado pela extensão dos NPs acima, ou seja, o conjunto das crianças, e por V o conjunto caracterizado pelas extensões dos VPs, ou seja, o conjunto dos indivíduos que choraram. Podemos dizer então que (39) é verdadeira se, e somente se, a interseção de N e V não for vazia, ou seja, se contiver pelo menos um elemento. Já (40) tem condições de verdades opostas, sendo verdadeira se, e somente se, a interseção de N e V for o conjunto vazio. Por fim, (41) é verdadeira se, e somente se, todo elemento de N também for um elemento de V, ou seja, se N for um subconjunto de V. Essas considerações podem ser convenientemente representadas em diagramas como os abaixo, em que o retângulo representa o conjunto de todos os indivíduos:

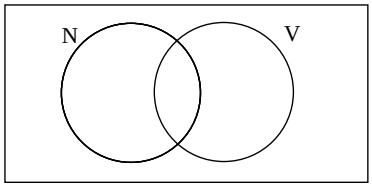


Figura 6.1: Algum: $\mathrm{N} \cap \mathrm{V} \neq \varnothing$

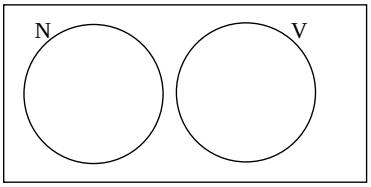


Figura 6.2: Nenhum: $N \cap V = \varnothing$

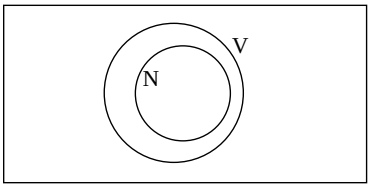


Figura 6.3: Todo: $N \subseteq V$

Note agora uma característica das interpretações das sentenças (39)-(41) relacionada a suas representações acima. Em todos os casos, para saber se a sentença é verdadeira, basta perguntar quantos membros do conjunto N são também membros do conjunto V. As respostas para os exemplos acima, claro, são: pelo menos um, nenhum e todos, respectivamente. Em nenhum dos casos, precisamos de informações sobre indivíduos que não pertencem ao conjunto N. Isto significa que as extensões dos determinantes acima podem ser caracterizadas levando-se em conta apenas os conjuntos N e $N \cap V$, sem levar em conta os membros de V que não pertencem a N ou os indivíduos que não pertencem nem a N nem a V:

(42) a. $[\![\text{algum}]\!](N)(V) = 1$ *sse* $N \cap V \neq \varnothing$
b. $[\![\text{nenhum}]\!](N)(V) = 1$ *sse* $N \cap V = \varnothing$
c. $[\![\text{todo}]\!](N)(V) = 1$ *sse* $N = N \cap V$

Mas se os elementos de V que não estão na interseção com N são mesmo irrelevantes, poderíamos trocar V em (42a-c) por $N \cap V$ sem perda semântica. Considere, então, o que acontece se substituirmos o conjunto V pelo conjunto $N \cap V$ nas condições acima:

(43) a. $[\![\text{algum}]\!](N)(N \cap V) = 1$ *sse* $N \cap (N \cap V) \neq \varnothing$
b. $[\![\text{nenhum}]\!](N)(N \cap V) = 1$ *sse* $N \cap (N \cap V) = \varnothing$
c. $[\![\text{todo}]\!](N)(N \cap V) = 1$ *sse* $N = N \cap (N \cap V)$

As condições de verdade acima parecem distintas das anteriores. Entretanto, pelas definições da teoria dos conjuntos, sabemos que, para quaisquer conjuntos A e B, vale a seguinte equivalência:

$A \cap (A \cap B) = A \cap B$

Isso nos permite simplificar as condições de verdade acima, reduzindo-as ao seguinte:

(44) a. $[\![\text{algum}]\!](N)(N \cap V) = 1$ *sse* $N \cap V \neq \varnothing$
b. $[\![\text{nenhum}]\!](N)(N \cap V) = 1$ *sse* $N \cap V = \varnothing$
c. $[\![\text{todo}]\!](N)(N \cap V) = 1$ *sse* $N = N \cap V$

Mas o que aparece à direita do conectivo *sse* nas condições em (44) é exatamente o que aparecia à direita desse conectivo nas condições em (42). Isto quer dizer que:

(45) a. $[\![\text{algum}]\!](N)(V) = [\![\text{algum}]\!](N)(N \cap V)$
b. $[\![\text{nenhum}]\!](N)(V) = [\![\text{nenhum}]\!](N)(N \cap V)$
c. $[\![\text{todo}]\!](N)(V) = [\![\text{nenhum}]\!](N)(N \cap V)$

Equivalências como essas parecem ser válidas para todos os determinantes das línguas naturais. Essas equivalências podem ser expressas em português por esquemas como os mostrados abaixo:

D NP VP ≡ D NP é um NP que VP

Concretamente:

Alguma criança chorou ≡ Alguma criança é uma criança que chorou

Nenhuma criança chorou ≡ Nenhuma criança é uma criança que chorou

Toda criança chorou ≡ Toda criança é uma criança que chorou

E são essas equivalências que estão no cerne da propriedade de conservatividade, que definimos a seguir:

Conservatividade
Para qualquer determinante D e quaisquer conjuntos A e B, dizemos que D é conservativo se, e somente se, $[\![D]\!](A)(B) = [\![D]\!](A)(A \cap B)$

Podemos, então, formular a seguinte hipótese para as línguas naturais:

Universal da Conservatividade
Todo determinante é conservativo.

A validade desse universal é algo sujeito a confirmação empírica. Como possível contraexemplo, considere o item *só* do português, que aparece em sentenças como (46) abaixo:

(46) Só crianças choram.

Note uma certa semelhança sintática dessa sentença com as que analisamos mais acima. A palavra *só* vem seguida de um NP e o constituinte resultante é o sujeito do VP que o segue. Parece que estamos, de fato, diante de um determinante. Pensemos agora nas condições de verdade de (46). Para que essa sentença seja verdadeira, é preciso que ninguém além das crianças chore, ou seja, que todo indivíduo que chore seja uma criança. Isto nos leva à seguinte entrada lexical para *só*:

$[\![\text{só}]\!] = \lambda F_{\langle et \rangle}.\lambda G_{\langle et \rangle}.\ \forall x[G'(x) \rightarrow F'(x)]$

Já na versão baseada em conjuntos que exploramos acima, podemos caracterizar as condições de verdade de (46) através do diagrama abaixo:

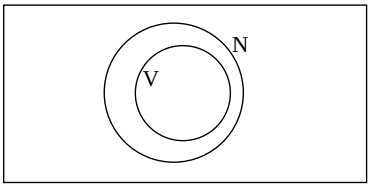


Figura 6.4: Só: V $\subseteq$ N

Note que, para sabermos se (46) é verdadeira, não basta perguntarmos quantos elementos de N pertencem a V, ou seja, quantas crianças choram. Precisamos

também de informação sobre a existência ou não de elementos que pertencem a V, mas não pertencem a N. Nesse aspecto, *só* difere dos determinantes que vimos anteriormente e essa diferença é fruto do fato de que *só* não é conservativo.

Para provar a conservatividade de *só*, basta observar a não equivalência entre *só crianças são crianças que choram* e *só crianças choram*, em oposição à equivalência entre, por exemplo, *toda criança é uma criança que chora* e *toda criança chora*. De um ponto de vista mais formal, basta olharmos para a caracterização de sua extensão na versão baseada em conjuntos:

$[\![\text{só}]\!](N)(V) = 1$ *sse* $V \subseteq N$

Note que o que aparece do lado direito do conectivo *sse* é uma contingência, ou seja, algo que pode ser verdadeiro ou falso dependendo da natureza de *N* e *V*. Considere agora o que acontece se substituirmos *V* por $N \cap V$:

$[\![\text{só}]\!](N)(N \cap V) = 1$ *sse* $(N \cap V) \subseteq N$

O que aparece agora após o conectivo *sse* é uma tautologia, já que, para quaisquer conjuntos *N* e *V*, a interseção de *N* e *V* é um subconjunto de N. A conclusão que tiramos disso é que as duas condições de verdade acima não são equivalentes, e que, portanto, *só* não é conservativo.

Essa conclusão bastaria para refutarmos o Universal da Conservatividade, não fosse o fato de que há sólidas evidências de que *só* não é um determinante. Ao contrário dos determinantes que discutimos anteriormente, *só* não precede apenas NPs, mas também DPs, como em *só o menino* ou *só algumas meninas*, VPs como em *João só dançou*, APs como em *João está só triste* e PPs, como em *João viaja só de primeira classe*. Isto leva a crer que *só* é na verdade um tipo de adjunto que não é seletivo em relação à categoria do constituinte ao qual se adjunge. Podemos assim manter viva a hipótese de que determinantes são universalmente conservativos.

Como já afirmamos acima, universais como esse são hipóteses a serem confirmadas empiricamente, e uma das linhas de pesquisa abertas pelo estudo formal das propriedades dos determinantes e dos quantificadores generalizados que eles originam é justamente a busca desses universais semânticos.

6.8 Quantificação e escopo

6.8.1 DPs quantificadores em posição de objeto

Todos os exemplos que analisamos até aqui neste capítulo continham sintagmas quantificadores na posição de sujeito. Tal escolha não se deu por acaso, e para entender o porquê de termos evitado outras posições sintáticas, vamos considerar um exemplo com um DP quantificador na posição de objeto direto de um verbo transitivo, como em (47):

(47) O João elogia todo professor.

A extensão do verbo *elogia* é de tipo $\langle e, et\rangle$ e a do DP *todo professor* de tipo $\langle et, t\rangle$. Não há, portanto, como utilizar aplicação funcional para obter a extensão do VP *elogia todo professor*. Outras regras composicionais que introduzimos, como abstração funcional ou conjunção funcional, também não nos servem. Essa situação contrasta com casos em que um DP aparece na posição de sujeito como em (48) abaixo, em que a extensão do VP *elogia o João* é de tipo $\langle et\rangle$ e pode servir de argumento para a extensão do DP sujeito, como já vimos anteriormente:

(48) Todo professor elogia o João.

Note que as condições de verdade tanto de (47) quanto de (48) são intuitivamente claras e facilmente representáveis na notação que estamos adotando:

(49) $[\![$O João elogia todo professor$]\!]$ = 1 *sse*
$\forall x[\textsc{professor}(x) \rightarrow \textsc{elogia}(\textit{joão},x)]$

(50) $[\![$Todo professor elogia o João$]\!]$ = 1 *sse*
$\forall x[\textsc{professor}(x) \rightarrow \textsc{elogia}(x,\textit{joão})]$

Em ambos os casos, temos o prefixo quantificador $\forall x$ seguido de uma afirmação contendo a variável x. Essa afirmação, uma fórmula lógica, é o ESCOPO de $\forall x$. A questão que nos resta é como obter composicionalmente essas condições de verdade no caso de (49), a começar pelo problema da incompatibilidade de tipos que acabamos de ver.

Logo adiante, veremos como reformular nosso sistema interpretativo e tornar exemplos como (47) interpretáveis, derivando suas condições de verdade sem perder nada do que já obtivemos para casos como (48). Antes, porém, vamos discutir dois fenômenos relacionados e que dizem respeito à maneira como DPs quantificadores interagem semanticamente entre si e com a negação.

6.8.2 Interação de escopo entre DPs

Vejamos agora casos em que mais de um DP quantificador aparece na mesma sentença, como em (51), por exemplo:

(51) Um parecerista anônimo revisará todo artigo que for submetido a esta revista.

A primeira coisa a se notar a respeito de (51) é que parecemos estar diante de uma sentença ambígua, que admite duas interpretações. A primeira interpretação afirma a existência de um certo parecerista que fará a revisão de todos os artigos que forem submetidos ao periódico em questão. De acordo com essa interpretação, para que (51) seja verdadeira, é preciso que um único parecerista seja responsável pela revisão de todos os artigos submetidos. Tal interpretação pode ser parafraseada por algo como: existe um parecerista *x*, tal que *x* revisará todo artigo submetido ao periódico. Ou ainda, e de maneira um pouco mais prolixa: existe um parecerista *x*, tal que, para todo artigo *y* submetido ao periódico, *x* revisará *y*. Usando a notação que empregamos nas seções anteriores, podemos representar abreviadamente essa interpretação da seguinte maneira:

(52) $\exists x[\textsc{parecerista}(x) \,\&\, \forall y[\textsc{artigo}(y) \rightarrow \textsc{revisará}(x,y)]]$

O que está escrito em (52) é que existe um indivíduo *x* que tem duas propriedades: *x* é parecerista e *x* revisará todo artigo em questão.

Antes de prosseguir, vamos olhar mais atentamente para a estrutura interna da representação formal em (52). Olhando de fora para dentro, notamos que (52) tem a forma $\exists x[\phi]$, em que ϕ é uma fórmula lógica. Essa fórmula é o escopo do quantificador $\exists x$ introduzido pelo determinante *um*. Note agora que no interior de ϕ está a contribuição do determinante *todo*, que podemos representar esquematicamente por $\forall y[\psi]$, em que ψ é também uma fórmula lógica. Em casos como esse, dizemos que o quantificador existencial introduzido pelo determinante *um* tem escopo sobre o quantificador universal introduzido pelo determinante *todo*, ou, vendo de ângulo oposto, que o quantificador universal introduzido pelo determinante *todo* tem escopo sob o quantificador existencial introduzido pelo determinante *um*. Pode-se dizer também que o DP sujeito tem escopo sobre o DP objeto.

Passemos agora para a segunda interpretação de (51), um pouco menos saliente que a primeira. De acordo com essa interpretação, todo artigo submetido ao periódico passará pela revisão de um parecerista, não necessariamente o mesmo.

Tal interpretação não exige a existência de um certo parecerista que revise todos os artigos em questão. Exige-se apenas que, para todo artigo y, exista um parecerista x, tal que x revise y. Formalmente, temos o seguinte:

(53) $\forall y[\textsc{artigo}(y) \rightarrow \exists x[\textsc{parecerista}(x) \,\&\, \textsc{revisará}(x,y)]]$

A representação em (53) nos diz que para todo y, se y é um artigo, então existe um x, tal que x é um parecerista e x revisará y. Note que, aqui, é o quantificador existencial $\exists$ que aparece no interior da formula introduzida pelo quantificador universal $\forall$. Dizemos nesse caso que o quantificador universal tem escopo sobre o existencial, ou ainda, que o DP *todo artigo submetido a este periódico* tem escopo sobre o DP *um parecerista*.

Analisando as interpretações de (52), deparamo-nos então com uma AMBIGUIDADE DE ESCOPO e queremos que nosso sistema interpretativo consiga ajudar a explicá-la.

6.8.3 Negação e escopo

Passemos, agora, a um caso em que um DP quantificador interage com um outro tipo de operador, a negação. Considere a sentença abaixo:

(54) João não elogia todo professor.

Na interpretação mais natural desta sentença, para que ela seja verdadeira é preciso que haja pelo menos um professor que o João não elogie. Dito de maneira um pouco mais prolixa, é falso que para todo professor x, o João elogia x. Podemos representar essas condições de verdade da seguinte forma:

(55) $\neg\forall x[\textsc{professor}(x) \rightarrow \textsc{elogia}(\textit{joão},x)]$

Para uma fórmula do tipo $\neg\phi$, em que ϕ também é uma fórmula, dizemos que ϕ corresponde ao escopo do operador negativo $\neg$. Para uma fórmula $\forall x[\phi]$, já vimos que ϕ é o escopo do quantificador universal $\forall$. Note que, em (55), o quantificador universal aparece no interior do escopo do operador negativo. Dizemos, então, que a negação tem escopo sobre o quantificador universal. Costuma-se dizer também que a negação tem escopo sobre o DP quantificador.

Já para uma interpretação de (54) em que o quantificador universal tivesse escopo sobre a negação, teríamos algo parafraseável como "para todo professor x, é falso que o João elogia x". Em termos formais:

(56) $\forall x[\textsc{professor}(x) \rightarrow \neg\textsc{elogia}(\textit{joão},x)]$

De acordo com (56), (54) é verdadeira se, e somente se, para qualquer professor *x*, não for verdade que João elogia *x*, ou seja se, e somente se, João não elogiar nenhum professor. Essa parece ser uma interpretação bem menos saliente para (54) e talvez só seja possível quando o DP quantificador estiver bastante enfatizado. De qualquer forma, precisamos ver como nosso sistema interpreta estruturas em que DPs quantificadores e negação estão presentes em uma mesma estrutura.

Nas duas próximas seções, vamos apresentar e discutir, uma de cada vez, duas estratégias para lidar com as questões levantadas nessa seção a respeito da interpretação de DPs quantificadores e seu escopo relativo a outros constituintes sintáticos. A primeira delas é eminentemente sintática, mantendo nosso sistema interpretativo intacto, mas operando transformações sobre a estrutura sintática das sentenças. Já a segunda, vai na direção oposta, sendo eminentemente semântica. De acordo com ela, a estrutura sintática relevante para sua interpretação é a estrutura superficial com DPs quantificadores ocupando as mesmas posições argumentais que DPs referenciais. Porém, os tipo semânticos atribuídos a alguns componentes da sentença são modificados através de regras de mudanças de tipos.

6.9 Quantificação e movimento

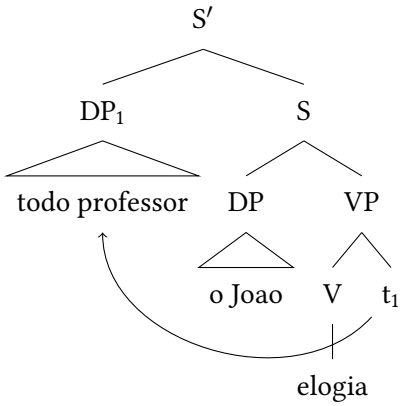
Voltemos ao caso dos DPs quantificadores em posição de objeto:

(57) O João elogia todo professor.

As condições de verdade dessa sentença exigem que, para todo indivíduo x, se x for professor, então x é elogiado pelo João. Na linha do que estamos vendo nesse capítulo, chegaríamos a esse resultado se fornecêssemos à extensão do determinante *todo* funções que caracterizassem o conjunto dos indivíduos que são professores e o conjunto do indivíduos que João elogia. Quanto ao primeiro, não há problema: trata-se da extensão do NP complemento de D. Mas e o segundo? Eis o que queremos:

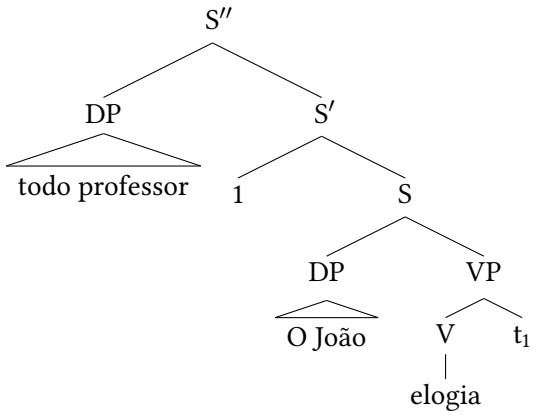
$\lambda x.$ ELOGIA(*joão,x*)

A primeira estratégia que veremos consiste em uma operação sintática de movimento. Nela, um DP quantificador se desloca para o topo de uma sentença, deixando para trás um vestígio. Esse DP movido e o seu vestígio recebem índices numéricos idênticos, como representado a seguir:



Essa operação sintática é conhecida como **alçamento de quantificador (AQ)** e tem como peculiaridade o fato de que não afeta a maneira como pronunciamos a sentença, apenas a maneira como a interpretamos. Em outras palavras, o resultado desta operação não é visível para o componente fonológico da gramática, apenas para o componente semântico.

A exemplo do que vimos quando discutimos a estrutura interna das orações relativas, vamos assumir, *a la* Heim & Kratzer (1998), um processo de transferência de índices na interface sintaxe-semântica, em que o índice do constituinte movido é transferido desse para o constituinte que for seu irmão, a ele se adjungindo.



Portanto, para a interpretação de (57), é a estrutura acima que o componente semântico recebe como *input*. Vejamos, agora, como nosso sistema interpreta esta estrutura. De acordo com o que vimos no capítulo anterior, quando introduzimos a regra de abstração funcional, temos que:

$[\![\text{S}']\!]^g = \lambda x.\ [\![\text{S}]\!]^{g[1\rightarrow x]}$

Como sabemos que (consulte o capítulo anterior se você tiver dúvidas):

$[\![\text{S}]\!]^{g[1\rightarrow x]} = 1$ *sse* ELOGIA(*joão,x*)

então,

$[\![\text{S}']\!]^g = \lambda x.$ ELOGIA(*joão,x*)

Isso é exatamente o que queríamos. Note que essa extensão é uma função de tipo $\langle et \rangle$, que é o que a extensão do DP movido toma como argumento. Podemos, então, utilizar aplicação funcional para obter a extensão de S'':

$[\![\text{S}'']\!]^g = [\![\text{DP}]\!]^g([\![\text{S}']\!]^g)$

$[\![\text{DP}]\!]^g = \lambda F.\ \forall x[\text{PROFESSOR}(x) \rightarrow\ F'(x)]$

$[\![\text{S}'']\!]^g = 1$ *sse* $\forall x[\text{PROFESSOR}(x) \rightarrow$ ELOGIA(*joão,x*)$]$

Note que, como a extensão acima não depende da atribuição *g* (essa não aparece à direita de *sse*), podemos, de acordo com a convenção que adotamos no capítulo anterior, escrever simplesmente:

$[\![\text{S}'']\!] = 1$ *sse* $\forall x[\text{PROFESSOR}(x) \rightarrow$ ELOGIA(*joão,x*)$]$

Essas eram exatamente as condições de verdade que queríamos derivar. Valendo-nos apenas dos princípios composicionais que já tínhamos à nossa disposição, em particular aplicação e abstração funcional, conseguimos derivar o significado de (57). O custo disso, claro, foi a postulação de uma operação sintática (AQ) sem reflexos fonológicos.

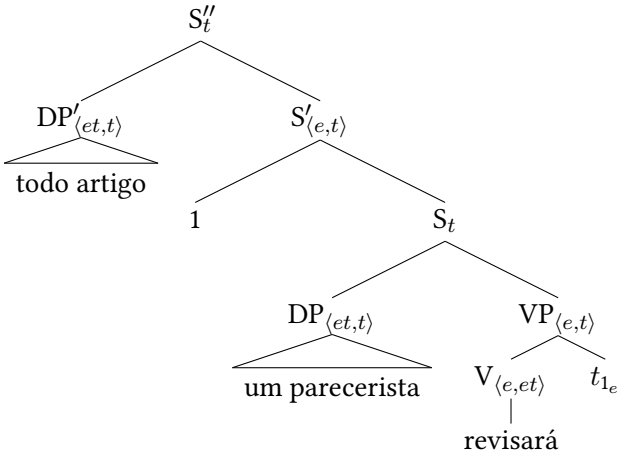
6.9.1 Movimento e escopo

A regra de alçamento de quantificadores move um DP quantificador para o topo da sentença, deixando em seu lugar de origem um vestígio. Vamos aplicá-la agora ao exemplo com dois DPs quantificadores que vimos anteriormente e que repetimos em (58):

(58) Um parecerista (anônimo) revisará todo artigo (que for submetido a esta revista).

O alvo de AQ será o DP *todo artigo*, o que resultará na estrutura abaixo (já assumindo a transferência de índice):

(59)



Nessa estrutura, foram anotados os tipos semânticos de todos os nós, de acordo com o previsto pelas entradas lexicais e as regras de aplicação funcional e abstração funcional. Note que não há incompatibilidades e que o nó raiz (S) é de tipo t, como deve ser. Vejamos, então, a interpretação que nosso sistema deriva para essa estrutura, procedendo de cima para baixo. Omitiremos alguns passos, deixando-os a cargo do leitor.

$$[\![S'']\!]^g = [\![DP']\!]^g([\![S']\!]^g) \tag{AF}$$

$$[\![DP']\!]^g = \lambda F.\ \forall x[\textsc{artigo}(x) \rightarrow F'(x)]$$

$$[\![S'']\!]^g = 1 \textit{ sse } \forall x[\textsc{artigo}(x) \rightarrow [\![S']\!]^g(x) = 1]$$

Para o nó S′, usamos abstração funcional:

$$[\![S']\!]^g = \lambda z.\ [\![S]\!]^{g[1\rightarrow z]}$$

Para evitar confusões nos processos de conversão-λ durante a derivação, usamos aqui a variável z, já que x e y aparecerão em outros momentos da derivação.

Trata-se, entretanto, da mesmíssima regra com a qual já estamos familiarizados. Para o nó S, voltamos a utilizar aplicação funcional:

$[\![\text{S}]\!]^{g[1\to z]} = [\![\text{DP}]\!]^{g[1\to z]}([\![\text{VP}]\!]^{g[1\to z]})$

$[\![\text{DP}]\!]^{g[1\to z]} = \lambda F.\ \exists\ y[\textsc{parecerista}(y)\ \&\ F'(y)]$

Por fim, para o nó VP, teremos, via aplicação funcional:

$[\![\text{VP}]\!]^{g[1\to z]} = [\![\text{V}]\!]^{g[1\to z]}([\![\text{t}_1]\!]^{g[1\to z]})$

$[\![\text{V}]\!]^{g[1\to z]} = \lambda v.\lambda w.\ \textsc{revisará}(w,v)$

$[\![\text{t}_1]\!]^{g[1\to z]} = g[1 \to z](1) = z$

Note o uso na representação da extensão de V das variáveis v e w, diferentes das já utilizadas mais acima. Aqui também o intuito é manter a clareza na exposição da derivação.

Tendo chegado à parte de baixo da estrutura, basta percorrermos o sentido inverso, efetuando as devidas conversões:

$[\![\text{VP}]\!]^{g[1\to z]} = \lambda w.\ \textsc{revisará}(w,z)$

$[\![\text{S}]\!]^{g[1\to z]} = 1$ *sse* $\exists y[\textsc{parecerista}(y)\ \&\ \textsc{revisará}(y,z)]$

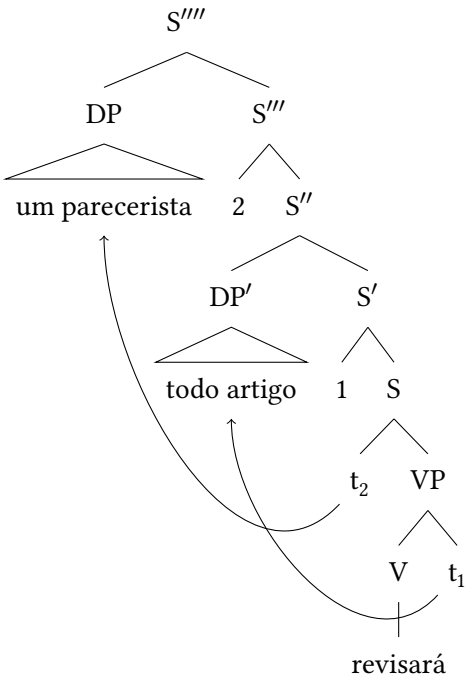
$[\![\text{S}']\!]^{g} = \lambda z.\ \exists y[\textsc{parecerista}(y)\ \&\ \textsc{revisará}(y,z)]$

$[\![\text{S}'']\!]^{g} = 1$ *sse* $\forall x[\textsc{artigo}(x) \to \exists y[\textsc{parecerista}(y)\ \&\ \textsc{revisará}(y,x)]]$

O que as condições acima requerem é que todo artigo seja revisado por um parecerista, não necessariamente o mesmo. Isso corresponde exatamente à leitura que discutimos na seção anterior em que o DP *todo artigo* tem escopo sobre o DP *um parecerista*. De fato, se compararmos as condições de verdade acima com a representação que atribuímos a essa leitura anteriormente em (53), notaremos que elas são idênticas, exceto pelas escolhas das variáveis, o que é irrelevante.

Derivamos, assim, uma das interpretações de (58). Mas, e a outra, que afirma a existência de um certo parecerista que revisará todos os artigos em questão? Note que, na estrutura que gerou as condições de verdade que acabamos de de-

rivar, o DP *todo artigo* aparece em uma posição hierarquicamente superior ao DP *um parecerista*, resultando na leitura em que o primeiro tem escopo sobre o segundo. Para obtermos a leitura em que *um parecerista* tem escopo sobre *todo artigo*, vamos aplicar AQ novamente, desta vez movendo o DP *um parecerista* por sobre o DP *todo artigo*. Isto resultará na seguinte estrutura:



Vejamos, então, que condições de verdade nosso sistema deriva para essa estrutura. Procure acompanhar atentamente cada passo da derivação:

$$[\![\text{S}'''']\!]^g = [\![\text{DP}]\!]^g([\![\text{S}''']\!]^g) \quad \text{(AF)}$$

$$[\![\text{S}''']\!]^g = \lambda x.\ [\![\text{S}'']\!]^{g[2\to x]} \quad \text{(Abs.F)}$$

$$[\![\text{S}'']\!]^{g[2\to x]} = [\![\text{DP}']\!]^{g[2\to x]}([\![\text{S}']\!]^{g[2\to x]}) \quad \text{(AF)}$$

$$[\![\text{S}']\!]^{g[2\to x]} = \lambda y.\ [\![\text{S}]\!]^{g[2\to x][1\to y]} \quad \text{(Abs.F)}$$

$$[\![\text{S}]\!]^{g[2\to x][1\to y]} = [\![\text{V}]\!]^{g[2\to x][1\to y]}([\![\text{t}_1]\!]^{g[2\to x][1\to y]})([\![\text{t}_2]\!]^{g[2\to x][1\to y]}) \quad \text{(Abs.F)}$$

$[\![\text{S}]\!]^{g[2\to x][1\to y]}$ = 1 *sse* REVISARÁ(x,y)

$[\![\text{S}']\!]^{g[2\to x]} = \lambda y.$ REVISARÁ(x,y)

$[\![\text{S}'']\!]^{g[2\to x]} = [\![\text{DP}']\!]^{g[2\to x]}(\lambda y.$ REVISARÁ(x,y))

$[\![\text{DP}']\!]^{g[2\to x]} = \lambda F.\ \forall y[\text{ARTIGO}(y) \to F'(y)]$

$[\![\text{S}'']\!]^{g[2\to x]}$ = 1 *sse* $\forall y[\text{ARTIGO}(y) \to$ REVISARÁ(x,y)]

$[\![\text{S}''']\!]^{g} = \lambda x_e.\ \forall y[\text{ARTIGO}(y) \to$ REVISARÁ(x,y)]

$[\![\text{S}'''']\!]^{g} = [\![\text{DP}]\!]^{g}(\lambda x.\ \forall y[\text{ARTIGO}(y) \to$ REVISARÁ(x,y)])

$[\![\text{DP}]\!]^{g} = \lambda F.\ \exists\ x[\text{PARECERISTA}(x)\ \&\ F'(x)]$

$[\![\text{S}'''']\!]^{g}$ = 1 *sse* $\exists\ x[\text{PARECERISTA}(x)\ \&\ \forall y[\text{ARTIGO}(y) \to$ REVISARÁ(x,y)]]

Se compararmos essas condições de verdade com a representação em (52) apresentada originalmente, veremos que elas são idênticas. Conseguimos, assim, captar a leitura de (58) em que *um parecerista* tem escopo sobre *todo artigo*.

Podemos concluir que a regra sintática AQ permite gerar duas estruturas sintáticas para (58) e que a interpretação de cada uma delas resulta em uma das leituras associadas a essa sentença. Explicamos, assim, a ambiguidade de (58), reduzindo-a a um caso de ambiguidade sintática, em que à mesma sentença (entendida como uma sequência de palavras) atribuímos duas estruturas distintas. O custo disso, voltamos a repetir, é a postulação de um nível de representação sintática abstrato e que não corresponde à estrutura superficial da sentença no que diz respeito à posição dos DPs quantificadores.

Cumpre notar também que não impusemos nenhuma restrição a AQ relacionada à natureza dos DPs quantificadores alvejados por ela. Dessa forma, prevemos que sempre que houver dois ou mais deles em uma mesma oração, haverá ambiguidade. Essa liberdade acaba sendo problemática na medida em que certas combinações de DPs não admitem múltiplas interpretações. Considere, por exemplo, o caso abaixo:

(60) Todo mundo resolveu menos de três questões.

Intuitivamente, a única leitura admissível para essa sentença é a de que para

todo x, x resolveu menos de três questões. Em outras palavras, não deve haver ninguém que tenha resolvido três ou mais questões. Essa é a interpretação em que o quantificador universal introduzido por *todo mundo* tem escopo largo, que inclui a contribuição semântica do DP objeto e que pode ser obtida alçando-se esse último para uma posição sintática subordinada à do DP sujeito:

(61) Todo mundo 1 [menos de três questões 2 [t_1 resolveu t_2]]

Considere, agora, a estrutura resultante do movimento do DP objeto para uma posição acima da do DP sujeito:

(62) menos de três questões 2 [todo mundo 1 [t_1 resolveu t_2]]

Nesse caso, teremos uma inversão de escopo e a interpretação resultante será a de que menos de três questões foram resolvidos por todo mundo. Mas essa não é uma interpretação que (60) tem. Pensemos um pouco mais a respeito. Imagine, por exemplo um cenário com 5 pessoas (p_1-p_5), cinco questões (q_1-q_5) e os fatos relevantes representados na tabela abaixo:

pessoas	**questões resolvidas**
p_1	q_1,q_2,q_3,q_4,q_5
p_2	q_1,q_2
p_3	q_1,q_2,q_4
p_4	q_1,q_2,q_5
p_5	q_1,q_2,q_3,q_4

Nesse cenário, é falso que todo mundo tenha resolvido menos de três questões. p_1, por exemplo resolveu todas as cinco. Mas é verdadeiro que menos de três questões tenham sido resolvidas por todo mundo (apenas duas, q_1 e q_2, foram). Se a inversão de escopo fosse possível, (60) deveria ser julgada verdadeira nessas circunstâncias. Como esse não é o caso, concluímos que essa inversão não é possível.

Além disso, parece haver diferenças translinguísticas em alguns casos. O português brasileiro, por exemplo, parece bastante rígido em relação à possibilidade de inversão de escopo. Mesmo em sentenças como as que vimos anteriormente, envolvendo quantificadores universal e existencial, tal inversão parece bem pouco saliente. Deixaremos essas observações como um alerta para o leitor. De alguma forma, AQ (ou qualquer operação que seja responsável pelo movimento dos DPs quantificadores) precisa ter seu papel regulado, com o movimento

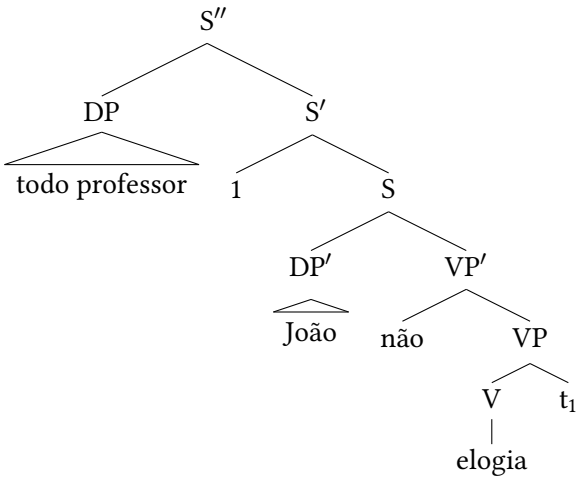
sintático em questão afetando de maneira distinta, diferentes tipos de DPs quantificadores. Remetemos o leitor a algumas das sugestões de leitura oferecidas ao final do capítulo.

6.9.2 Negação e alçamento de quantificadores

Considere novamente a sentença abaixo:

(63) João não elogia todo professor.

Conforme já discutimos, a interpretação mais natural dessa sentença exige que seja falso que João elogie todo professor. Vejamos se nosso sistema consegue derivar as condições de verdade almejadas. Já sabemos, que para interpretar um DP quantificador na posição de objeto direto, precisamos movê-lo para o topo da sentença. Isso nos fornece a seguinte estrutura:



Calculemos as condições de verdade dessa estrutura:

$[\![\text{S}'']\!]^g = [\![\text{DP}]\!]^g([\![\text{S}']\!]^g)$

$[\![\text{S}']\!]^g = \lambda x_e.\ [\![\text{S}]\!]^{g[1\to x]}$

$[\![\text{S}]\!]^{g[1\to x]} = [\![\text{VP}']\!]^{g[1\to x]}([\![\text{DP}']\!])^{g[1\to x]}$

$[\![\text{VP}']\!]^{g[1\to x]} = [\![\text{não}]\!]^{g[1\to x]}([\![\text{VP}]\!]^{g[1\to x]})$

$[\![\text{VP}]\!]^{g[1\to x]} = [\![\text{V}]\!]^{g[1\to x]}([\![\text{t}_1]\!]^{g[1\to x]})$

$[\![\text{V}]\!]^{g[1\to x]} = \lambda x.\lambda y.\ \textsc{elogia}(y,x)$

$[\![\text{t}_1]\!]^{g[1\to x]} = x$

$[\![\text{VP}]\!]^{g[1\to x]} = \lambda y.\ \textsc{elogia}(y,x)$

$[\![\text{não}]\!]^{g[1\to x]} = \lambda F.\ \lambda z.\ \neg F'(z)$

$[\![\text{VP}']\!]^{g[1\to x]} = \lambda z.\ \neg\textsc{elogia}(z,x)$

$[\![\text{DP}']\!]^{g[1\to x]} = \textit{joão}$

$[\![\text{S}]\!]^{g[1\to x]} = 1$ *sse* $\neg\textsc{elogia}(\textit{joão},x)$

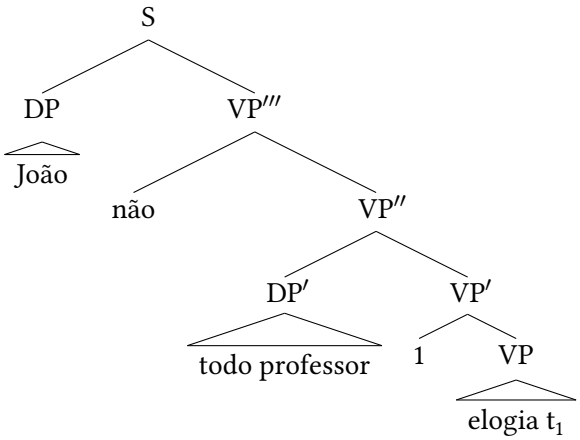
$[\![\text{S}']\!]^{g} = \lambda x.\ \neg\textsc{elogia}(\textit{joão},x)$

$[\![\text{DP}]\!]^{g} = \lambda F.\ \forall x[\textsc{professor}(x) \to F'(x)]$

$[\![\text{S}'']\!]^{g} = 1$ *sse* $\forall x[\textsc{professor}(x) \to \neg\textsc{elogia}(\textit{joão},x)]$

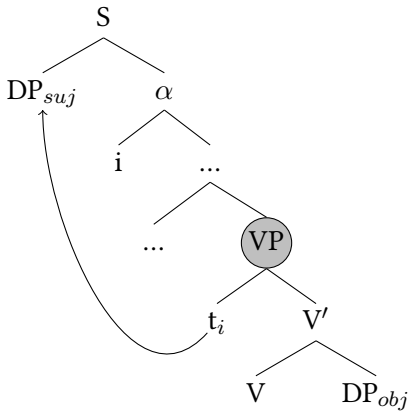
O que essas condições de verdade requerem é que, para todo professor *x*, o João não elogie *x*, ou seja, que o João não elogie nenhum professor. Conforme já salientamos, essa interpretação, se é que existe, é bem menos saliente que a interpretação de acordo com a qual nega-se apenas que o João elogie todo e qualquer professor. O problema com as condições que acabamos de derivar é que, de acordo com elas, o quantificador universal tem escopo sobre a negação, enquanto o que queríamos era que a negação tivesse escopo sobre o quantificador universal.

Mas como obter a interpretação de (63) que desejamos? Dado o que já vimos sobre interação de escopo na seção anterior, a resposta parece simples: basta mover o DP *todo professor* para uma posição abaixo da negação. Vejamos se isto realmente funciona. A estrutura que obteríamos seria a seguinte:

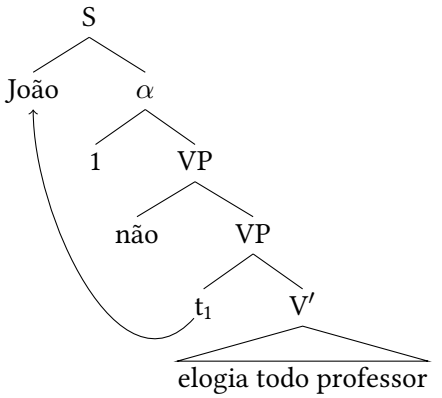


O problema é que essa estrutura não é interpretável. A extensão de VP é de tipo $\langle e, t\rangle$. Com isso a extensão de VP′ será de tipo $\langle e, et\rangle$ (por quê?). Como a extensão de *todo professor* é de tipo $\langle et, t\rangle$, não há como derivar a extensão de VP″ e, por conseguinte, de nenhum nó acima dele, incluindo o nó-raiz S. Note que, para tornar VP″ interpretável, bastaria que a extensão de VP′ fosse de tipo $\langle e, t\rangle$ e que dessa forma pudesse servir de argumento para a extensão de *todo professor*. Mas, para isso, a extensão de VP precisaria ser de tipo *t*, o que exigiria que o sujeito fosse interpretado no interior de VP, saturando assim a extensão do verbo *elogiar*.

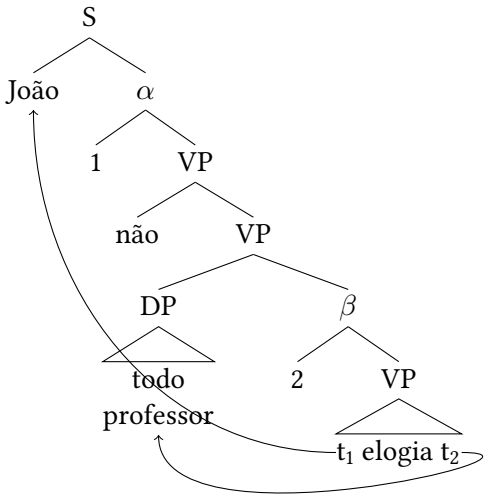
Para implementar essa possibilidade, vamos nos valer daquilo que, dentro do quadro teórico conhecido como Teoria da regência e ligação, é chamado de hipótese do sujeito interno a VP (ver referências ao final do capítulo). De acordo com essa hipótese, o sujeito de uma sentença é gerado no interior de VP, movendo-se depois para a posição que ocupa na estrutura superficial da sentença. Assim como nos outros casos envolvendo movimento que já estudamos, vamos assumir que, ao se mover, o DP sujeito recebe um índice e deixa em sua posição de origem um vestígio coindexado. Após o processo da transferência de índice, obtém-se então a seguinte estrutura:



Vamos aplicar essas ideias ao nosso exemplo. Primeiramente, movemos o sujeito para sua posição superficial:



Em seguida, movemos o DP objeto para a posição imediatamente abaixo da negação:



Eis os principais passos da derivação das condições de verdade dessa estrutura:

$[\![\text{S}]\!]^g = [\![\text{1 não todo professor 2 VP}]\!]^g(\textit{joão})$

$[\![\text{1 não todo professor 2 VP}]\!]^g = \lambda x.\ [\![\text{não todo professor 2 VP }]\!]^{g[1\to x]}$

$[\![\text{não todo professor 2 VP}]\!]^{g[1\to x]} = [\![\text{não}]\!]^{g[1\to x]}([\![\text{todo professor 2 VP}]\!])^{g[1\to x]}$

$[\![\text{todo professor 2 VP}]\!]^{g[1\to x]} = [\![\text{todo professor}]\!]^{g[1\to x]}(\text{2 VP})^{g[1\to x]}$

$[\![\text{2 VP}]\!]^{g[1\to x]} = \lambda y.\ [\![\text{VP}]\!]^{g[1\to x][2\to y]}$

$[\![\text{VP}]\!]^{g[1\to x][2\to y]} = 1$ *sse* $\textsc{elogia}(x,y)$

Note que a extensão de VP é de tipo *t*. Efetuando-se as devidas substituições e conversões-λ, obteremos o seguinte:

$[\![\text{2 VP}]\!]^{g[1\to x]} = \lambda y.\ \textsc{elogia}(x,y)$

$[\![\text{todo professor 2 VP}]\!]^{g[1\to x]} = 1$ *sse* $\forall y[\textsc{professor}(y) \to \textsc{elogia}(x,y)]$

Essa última extensão é de tipo t. Portanto, para a negação, deve-se usar a versão

de tipo $\langle t, t\rangle$, de acordo com a entrada lexical que propusemos no capítulo 3:

$[\![\text{não}]\!] = \lambda p.\ p = 0$

Prosseguindo, teremos:

$[\![\text{não todo professor 2 VP}]\!]^{g[1\to x]} = (\lambda p.\ p = 0)([\![\text{todo professor 2 VP}]\!])^{g[1\to x]}$

$[\![\text{não todo professor 2 VP}]\!]^{g[1\to x]} = 1$ *sse* $[\![\text{todo professor 2 VP}]\!]^{g[1\to x]} = 0$

$[\![\text{não todo professor 2 VP}]\!]^{g[1\to x]} = 1$ *sse* $\neg\forall y[\textsc{professor}(y) \to \textsc{elogia}(x,y)]$

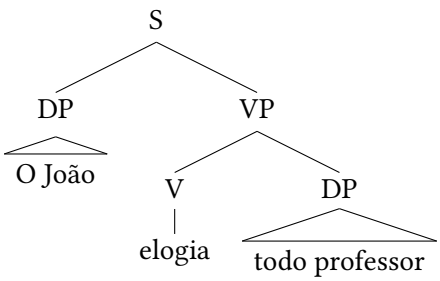
$[\![\text{1 não todo professor 2 VP}]\!]^{g[1\to x]} = \lambda x.\ \neg\forall y[\textsc{professor}(y) \to \textsc{elogia}(x,y)]$

$[\![\text{S}]\!] = 1$ *sse* $\neg\forall y[\textsc{professor}(y) \to \textsc{elogia}(\textit{joão},y)]$

Essas são exatamente as condições que queríamos derivar, com a negação tendo escopo sobre o quantificador universal. Novamente, valemo-nos de operações de movimento sintático que preparam, por assim dizer, a estrutura sintática para a interpretação baseada apenas em aplicação e abstração funcional, sem necessidade de rever os tipos semânticos dos constituintes envolvidos.

6.10 Quantificação e mudança de tipos

Vejamos, agora, uma segunda estratégia para a interpretação de sentenças com DPs quantificadores em diversas posições sintáticas. Como anunciamos mais acima, essa estratégia mantém a estrutura sintática superficial da sentença inalterada. No caso de *João elogia todo professor*, por exemplo, o que o componente semântico recebe para interpretar é a estrutura abaixo:



Para evitarmos a incompatibilidade de tipos que já detectamos entre as extensões de V e do DP objeto, precisamos alterar pelo menos uma dessas extensões, ou então criarmos um novo princípio composicional. Vamos aqui implementar a primeira possibilidade, alterando o tipo semântico dos verbos.

Já notamos no início deste capítulo que, do ponto de vista sintático, existe uma semelhança na distribuição de expressões referenciais como nomes próprios e a distribuição de DPs quantificadores, ambos ocupando as mesmas posições junto aos predicados a eles relacionados. A ideia por trás da implementação que vamos desenvolver aqui é a de que as extensões dos predicados estão em sintonia com esses fatos e que regras semânticas podem alterar o tipo dessas extensões, de modo a torná-las aptas a tomar não só indivíduos (tipo e) mas também quantificadores generalizados (tipo $\langle et, t\rangle$) como argumentos. De maneira mais concreta, isso significa que há regras que relacionam sistematicamente predicados de indivíduos e predicados de quantificadores generalizados. Vamos aqui tomar os primeiros como básicos e derivar a extensão dos últimos elevando o tipo semântico de seus argumentos de e para $\langle et, t\rangle$ quando necessário. Chamaremos essa regra de **Elevação de Argumento (EA)**.

Elevação de Argumento (EA)
Seja X um constituinte cuja extensão α é de tipo $\langle a_1, ...\langle a_k, ...\langle a_n, t\rangle ...\rangle\rangle$, com $n \geq 1$ e $a_k = e$. Mude a extensão de X de α para α', mudando a_k de e para $\langle et, t\rangle$ e definindo α' da seguinte forma:

$\alpha' = \lambda x_1 ... \lambda Q_{\langle et,t\rangle} ... \lambda x_n .\ Q(\lambda x_k .\ \alpha(x_1)...(x_k)...(x_n))$

A regra acima aplica-se a funções com *n* argumentos $x_1, ..., x_k, ..., x_n$, sendo o k-ésimo desses argumentos (x_k) de tipo *e*. A regra modifica o tipo desse k-ésimo argumento, passando-o de e para $\langle et, t\rangle$. É esse novo argumento *Q* que será fornecido por um DP quantificador ocupando uma posição argumental. Vejamos um exemplo concreto com o verbo transitivo *elogiar*. A extensão desse verbo é uma função de tipo $\langle e, et\rangle$, que toma dois argumentos de tipo *e*. No esquema acima, portanto, $n = 2$ e tanto a_1 quanto a_2 são iguais a *e*. Vamos aplicar EA, tomando *k* como sendo igual a 1, e portanto, x_k como sendo o primeiro argumento, que corresponde à posição de objeto direto. Esse argumento terá seu tipo elevado de e para $\langle et, t\rangle$. A nova extensão do verbo será, portanto, de tipo $\langle\langle et, t\rangle, et\rangle$. Lembremos que a extensão original do verbo *elogiar*, de tipo $\langle e, et\rangle$, é:

$[\![\text{elogia}_1]\!] = \lambda x.\lambda y.\ \textsc{elogia}(y,x)$

Marcamos o verbo com um subscrito para diferenciar a extensão original da nova extensão que obteremos, aplicando a mudança de tipos especificada acima:

$[\![\text{elogia}_2]\!] = \lambda Q_{\langle et,t\rangle}.\lambda y.\ Q(\lambda x.\ [\![\text{elogia}_1]\!](x)(y))$

Efetuando as conversões lambda, teremos:

$[\![\text{elogia}_2]\!] = \lambda Q_{\langle et,t\rangle}.\lambda y.\ Q(\lambda x.\ \textsc{elogia}(y,x))$

Com essa nova extensão, o verbo *elogiar* está apto a se combinar com um DP quantificador na posição de objeto, resultando em um VP cuja extensão terá o tipo $\langle e,t\rangle$, conforme se pode ver na derivação abaixo para a sentença *João elogia todo professor*. Para o DP objeto, temos:

$[\![\text{DP}]\!] = \lambda F.\ \forall z[\textsc{professor}(z) \rightarrow\ F'(z)]$

Sendo a nova extensão do verbo de tipo $\langle\langle et,t\rangle,et\rangle$, podemos utilizar aplicação funcional para obter a extensão do VP *elogia todo professor*:

$[\![\text{VP}]\!] = [\![\text{elogia}_2]\!]([\![\text{DP}]\!])$

$[\![\text{VP}]\!] = (\lambda Q_{\langle et,t\rangle}.\lambda y.\ Q(\lambda x.\ \textsc{elogia}(y,x)))([\![\text{DP}]\!])$

Basta, agora, efetuar uma série de substituições e conversões-λ:

$[\![\text{VP}]\!] = \lambda y.\ [\![\text{DP}]\!](\lambda x.\ \textsc{elogia}(y,x))$

$[\![\text{VP}]\!] = \lambda y.\ (\lambda F.\ \forall z[\textsc{professor}(z) \rightarrow F'(z)])(\lambda x.\ \textsc{elogia}(y,x))$

$[\![\text{VP}]\!] = \lambda y.\ \ \forall z[\textsc{professor}(z) \rightarrow\ (\lambda x.\ \textsc{elogia}(y,x))(z)]$

$[\![\text{VP}]\!] = \lambda y.\ \forall z[\textsc{professor}(z) \rightarrow\ \textsc{elogia}(y,z)]$

Essa extensão de VP é uma função que toma como argumento um indivíduo y e retorna o valor de verdade 1 se, e somente se, y elogia todo professor. Aplicando-a à extensão do DP sujeito *João*, temos:

$[\![\text{S}]\!] = [\![\text{VP}]\!]([\![\text{DP}]\!])$

$[\![\text{DP}]\!] = \textit{joão}$

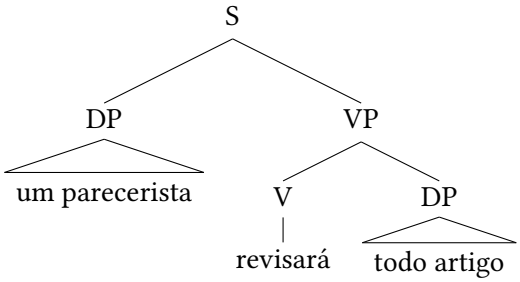
$[\![\text{S}]\!] = 1$ *sse* $\forall z[\textsc{professor}(z) \rightarrow \textsc{elogia}(\textit{joão},z)]$

Estas são, de fato, as condições de verdade que desejávamos. Ao contrário do que ocorreu com a primeira estratégia baseada em movimento, desta vez não lançamos mão de nenhuma operação sintática nos moldes de AQ. O custo para mantermos nossa sintaxe intacta foi a introdução de uma nova regra de mudança de tipos.

Não é óbvio como decidir entre as estratégias descritas acima, nem do ponto de vista empírico nem do ponto de vista metodológico. Ambas, no intuito de dar conta da interpretação de sentenças com quantificadores na posição de objeto, complicam, por assim dizer, um dos componentes da gramática. A primeira, complica o componente sintático, e a segunda, o semântico. Deixaremos essa escolha em aberto.

6.10.1 Mudança de tipos e escopo

Vejamos, agora, como a abordagem acima, que dispensa o uso de movimento sintático, lida com os fenômenos da ambiguidade de escopo entre DPs e da interação entre a negação e DPs quantificadores. Voltemos ao exemplo *Um parecerista anônimo revisará todo artigo que for submetido a este periódico* e sua estrutura superficial abreviada, dada abaixo:



Na seção anterior, aplicamos a regra de mudança de tipos que chamamos de Elevação de Argumento (EA) a um verbo transitivo, mudando o tipo semântico do primeiro argumento de sua extensão de e para $\langle et, t\rangle$. Isso possibilitou à nova extensão do verbo combinar-se diretamente com a extensão do seu objeto direto

quantificado via aplicação funcional, resultando em uma extensão de tipo $\langle e, t\rangle$ para o sintagma verbal VP. Se adotarmos o mesmo procedimento com o verbo *revisará* no exemplo acima, obteremos resultado análogo:

$[\![\text{revisará}_2]\!] = \lambda Q_{\langle et,t\rangle}.\lambda y.\ Q(\lambda x.\ \textsc{revisará}(y,x))$

Essa extensão tomará a extensão do DP objeto como argumento, via aplicação funcional:

$[\![\text{VP}]\!] = [\![\text{revisará}_2]\!]([\![\text{todo artigo}]\!])$

$[\![\text{todo artigo}]\!] = \lambda F_{\langle et\rangle}.\ \forall z[\textsc{artigo}(z)\ \rightarrow\ F'(z)]$

$[\![\text{VP}]\!] = \lambda y.\ \forall z[\textsc{artigo}(z) \rightarrow \textsc{revisará}(y,z)]$

Como a extensão do DP sujeito é de tipo $\langle et, t\rangle$, ela pode tomar a extensão do VP como argumento. Utilizando, portanto, aplicação funcional, obtemos as condições de verdade para a sentença S:

$[\![\text{S}]\!] = [\![\text{um parecerista}]\!]([\![\text{VP}]\!])$

$[\![\text{um parecerista}]\!] = \lambda F.\ \exists x[\textsc{parecerista}(x)\ \&\ F'(x)]$

$[\![\text{S}]\!] = 1$ *sse* $\exists x[\textsc{parecerista}(x)\ \&\ \forall z[\textsc{artigo}(z) \rightarrow \textsc{revisará}(x,z)]]$

Como se pode notar nessa representação das condições de verdade, o quantificador existencial tem escopo sobre o quantificador universal, e a interpretação resultante é a de que um mesmo parecerista revisará todos os artigos em questão. Falta-nos agora obter a outra interpretação, que traz uma inversão de escopo, ou seja, o quantificador universal introduzido pelo DP objeto passa a ter escopo sobre o quantificador existencial introduzido pelo DP sujeito. Para tanto, podemos continuar nos valendo de EA. Desta vez, porém, elevaremos inicialmente o tipo semântico do segundo argumento (o argumento externo) do verbo (k será, portanto, igual a 2 no esquema de EA):

$[\![\text{revisará}_1]\!] = \lambda x.\lambda y.\ \textsc{revisará}(y,x)$

$[\![\text{revisará}_2]\!] = \lambda x.\lambda Q_{\langle et,t\rangle}.\ Q(\lambda y.\ \textsc{revisará}(y,x))$

Note que essa aplicação de EA possibilita à extensão do verbo tomar um quantificador generalizado como segundo argumento. Precisamos agora permitir que a mesma tome um outro quantificador como seu primeiro argumento. Para isso podemos aplicar EA novamente. desta vez, entretanto, ela incidirá sobre a nova extensão de *revisará* que acabamos de obter, elevando o tipo semântico do primeiro argumento, de e para $\langle et, t\rangle$. Eis o que obtemos:

$[\![\text{revisará}_3]\!] = \lambda Q'_{\langle et,t\rangle}.\lambda Q_{\langle et,t\rangle}.\ Q'(\lambda x.\ [\![\text{revisará}_2]\!](x)(Q))$

$[\![\text{revisará}_3]\!] = \lambda Q'_{\langle et,t\rangle}.\lambda Q_{\langle et,t\rangle}.\ Q'(\lambda x.\ Q(\lambda y.\ \textsc{revisará}(y,x)))$

Temos, assim, uma nova extensão para o verbo *revisará* que corresponde a uma função que toma dois argumentos de tipo $\langle et, t\rangle$. Com isso, podemos chegar às condições de verdade de nossa estrutura, valendo-nos apenas de aplicação funcional. Comecemos com VP (preste bastante atenção nas conversões-λ):

$[\![\text{VP}]\!] = [\![\text{revisará}_3]\!]([\![\text{todo artigo}]\!])$

$[\![\text{VP}]\!] = (\lambda Q'_{\langle et,t\rangle}.\lambda Q_{\langle et,t\rangle}.\ Q'(\lambda x.\ Q(\lambda y.\ \textsc{revisará}(y,x))))([\![\text{todo artigo}]\!])$

$[\![\text{VP}]\!] = \lambda Q.\ [\![\text{todo artigo}]\!](\lambda x.\ Q(\lambda y.\ \textsc{revisará}(y,x)))$

$[\![\text{VP}]\!] = \lambda Q.\ \forall z[\textsc{artigo}(z)\ \rightarrow\ Q(\lambda y.\ \textsc{revisará}(y,x))]$

Para a extensão de S, temos:

$[\![\text{S}]\!] = [\![\text{VP}]\!]([\![\text{um parecerista}]\!])$

$[\![\text{S}]\!] = (\lambda Q_{\langle et,t\rangle}.\ \forall z[\textsc{artigo}(z)\ \rightarrow\ Q(\lambda y.\ \textsc{revisará}(y,x))])([\![\text{um parecerista}]\!])$

$[\![\text{S}]\!] = 1$ *sse* $\forall z[\textsc{artigo}(z)\ \rightarrow\ \exists y[\textsc{parecerista}(y)\ \&\ \textsc{revisará}(y,z)]]$

Como se pode notar, essas são as condições de verdade de acordo com as quais todo artigo será revisado por um parecerista, não necessariamente o mesmo.

Conseguimos assim captar a ambiguidade de escopo que almejávamos, utilizando apenas a regra EA de mudança de tipos. Como mostramos acima, a ordem de elevação dos tipos dos argumentos da extensão do verbo determinou as di-

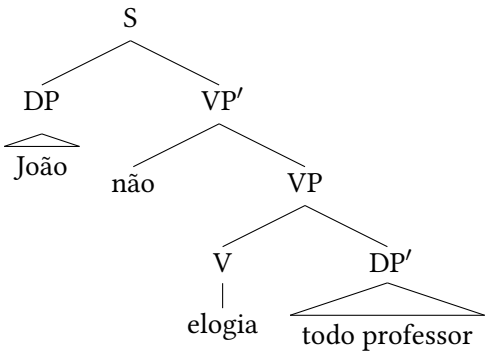
ferentes ordens de escopo. Quando o primeiro argumento (correspondente ao objeto direto) foi elevado inicialmente, o resultado foi o DP objeto direto com escopo mais estreito. Quando esse primeiro argumento foi elevado por último, o resultado foi o DP objeto direto com escopo mais largo e sobre o DP sujeito (inversão de escopo).

Por fim, cabe aqui o mesmo alerta que fizemos ao término da seção sobre AQ e ambiguidade de escopo. Da forma como foi definida, também EA é insensível à natureza dos DPs em questão, prevendo ambiguidade sempre que houver mais de um deles atrelados a um mesmo predicado. Como salientamos lá, essa liberdade na aplicação da regra leva a efeitos indesejados de sobregeração de interpretações.

6.10.2 Mudança de tipos, negação e escopo

Passemos, por fim, à interação entre a negação e DPs quantificadores, considerando novamente a sentença em (64) e a estrutura logo abaixo:

(64) João não elogia todo professor.



Vamos proceder de baixo para cima na derivação das condições de verdade desta estrutura, começando pela extensão do verbo *elogiar* obtida através de EA via elevação do tipo de seu primeiro argumento, conforme já discutido nas seções acima:

$[\![\text{elogia}_2]\!] = \lambda Q.\lambda z.\ Q(\lambda y.\ \textsc{elogia}(z,y))$

$[\![\text{todo professor}]\!] = \lambda F.\ \forall x[\textsc{professor}(x) \rightarrow\ F'(x)]$

$[\![\text{VP}]\!] = [\![\text{V}]\!]([\![\text{DP}']\!])$

$[\![\text{VP}]\!] = (\lambda Q.\lambda z.\ Q(\lambda y.\ \textsc{elogia}(z,y)))(\text{todo professor})$

$[\![\text{VP}]\!] = \lambda z.\ [\![\text{todo professor}]\!](\lambda y.\ \textsc{elogia}(z,y))$

$[\![\text{VP}]\!] = \lambda z.\ \forall x[\textsc{professor}(x) \rightarrow\ \textsc{elogia}(z,x)]$

Note que a extensão de VP é de tipo $\langle e, t\rangle$. Este é um tipo booleano (termina em t) e, portanto, podemos combinar a extensão de VP com a extensão da negação de tipo $\langle e, t\rangle, \langle e, t\rangle$, obtida através do esquema correspondente a sua entrada lexical que apresentamos no capítulo 3 (estude novamente a seção sobre negação naquele capítulo):

$[\![\text{não}]\!] = \lambda F_{\langle e,t\rangle}.\lambda y.\ \neg F'(y)$

$[\![\text{VP}']\!] = [\![\text{não}]\!]([\![\text{VP}]\!])$

$[\![\text{VP}']\!] = \lambda y.\ [\![\text{VP}]\!](y) = 0$

$[\![\text{VP}']\!] = \lambda y.\ \neg\forall x[\textsc{professor}(x) \rightarrow\ \textsc{elogia}(y,x)]$

Aplicando a extensão VP′ à extensão do DP sujeito, obtemos:

$[\![\text{S}]\!] = [\![\text{VP}']\!]([\![\text{DP}]\!])$

$[\![\text{DP}]\!] = \textit{joão}$

$[\![\text{S}]\!] = 1$ *sse* $\neg\forall x[\textsc{professor}(x) \rightarrow\ \textsc{elogia}(\textit{joão},x)]$

Como se pode notar, essas são as condições que queríamos derivar, com a negação tendo escopo sobre o quantificador universal. (In)felizmente (ver exercício 5), não conseguimos derivar, dentro do sistema baseado em mudança de tipos que estamos discutindo nesse capítulo, a interpretação em que o DP objeto tem escopo sobre a negação (mas ver sugestões de leitura abaixo).

Encerramos, assim, a versão de nosso sistema interpretativo que dispensa o uso de movimento sintático e que, através do uso de regras de mudança de tipos, é capaz de lidar com DPs quantificadores na posição de objeto, ambiguidade de

escopo e certas interações entre negação e DPs quantificadores.

Sugestões de leitura

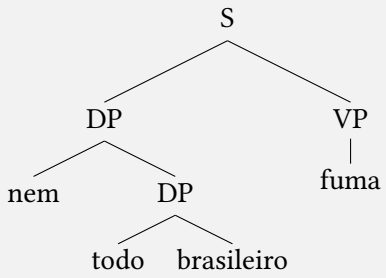
Para uma apresentação dos quantificadores existencial e universal no âmbito da lógica de predicados clássica, ver Gamut (1991) e Mortari (2016). Para o desenvolvimento no âmbito linguístico da ideia dos quantificadores generalizados, ver Barwise & Cooper (1981). Para um estudo aprofundado da quantificação, ver Peters & Westerståhl (2008). Para uma apresentação panorâmica sobre a quantificação nas línguas naturais, ver Szabolcsi (2010). Para saber mais sobre a interação entre pluralidade e quantificação, ver Winter (2001). Sobre numerais e DPs cardinais, ver Geurts (2006), Geurts & Nouwen (2007) e Krifka (1999). A regra de alçamento de quantificadores foi proposta por Robert May (ver May (1977; 1985)). Para uma discussão de vários tópicos ligados à sintaxe e à semântica dos DPs quantificadores na linhagem chomskyana, ver Hornstein (1994). Para uma teoria em que o movimento dos DPs quantificadores é sensível à sua natureza, ver Beghelli & Stowell (1997). Para discussões introdutórias à ideia do sujeito interno a VP, ver Carnie (2013) e Haegeman (1994). Algumas regras de mudanças de tipos relacionadas às que vimos neste capítulo são apresentadas e discutidas em Partee (1986) e Partee & Rooth (1983). A regra de elevação de argumentos foi proposta em Hendriks (1993), que também propôs uma outra regra chamada de *value rasing*, capaz de lidar com a interação entre negação e DPs que vimos na última seção desse capítulo.

Exercícios

I. Assuma que a sentença em (1) tenha a estrutura em (2):

(1) Nem todo brasileiro fuma.

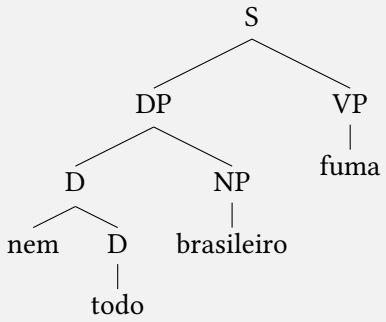
(2)



Proponha uma extensão para *nem* condizente com as condições de verdade da sentença acima.

Considere agora a estrutura em (3) e proponha uma nova entrada lexical para *nem* que a torne interpretável.

(3)



Qual a relação dessas extensões com a entrada lexical polimórfica da negação que vimos no capítulo 3?

II. Calcule, passo a passo, as condições de verdade da sentença abaixo:

(1) Mais de 5 alunos e menos de três professores chegaram.

III. Quais as condições de verdade que o sistema que desenvolvemos até aqui atribui à sentença abaixo?

(1) João não viu ninguém.

Essas condições de verdade são adequadas? Por quê?

IV. Imagine que o português possuísse dois determinantes — *toneg* e *alneg* — cujas extensões fossem as seguintes:

$[\![\text{toneg}]\!] = \lambda F.\ \lambda G.\ \forall x[\neg F'(x) \rightarrow G'(x)]$

$[\![\text{alneg}]\!] = \lambda F.\ \lambda G.\ \forall x[F'(x) \rightarrow \neg G'(x)]$

Nesse caso, quais seriam as condições de verdade das sentenças abaixo?

(1) Toneg brasileiro fuma.

(2) Alneg brasileiro fuma.

Diga se *toneg* e *alneg* são ou não conservativos, justificando sua resposta.

V. Mostre que EA, a regra de mudança de tipos apresentada neste capítulo, não é capaz de gerar para a sentença (1) abaixo a leitura em que o DP *todo professor* tem escopo sobre a negação.

(1) João não elogiou todo professor.

VI. Considere a sentença abaixo:

(1) A mãe de nenhum aluno reclamou.

Para muitos falantes, essa sentença significa que nenhuma mãe de aluno reclamou.

(i) Mostre que mover o DP *nenhum aluno* para o topo da sentença gera essa interpretação.

(ii) Mostre que, mesmo se aplicarmos a regra de elevação de argumentos (EA) de modo que ela também abarque nomes transitivos como *mãe*, a interpretação resultante não é a desejada.

7 Ligação

Este último capítulo é dedicado à LIGAÇÃO PRONOMINAL e à sua relação com a correferência e com o movimento sintático. Discutiremos também a possibilidade de uma semântica sem variáveis, em que a implementação da ligação se dá sem atribuições nem índices.

7.1 Pronomes livres e ligados

No capítulo 4, introduzimos a noção de atribuição e discutimos seu papel na codificação de informação contextual relacionada a referentes tornados salientes no contexto. A partir disso, assumimos que as extensões dos pronomes pessoais variam de contexto a contexto e implementamos essa ideia através da atribuição de índices a esses pronomes e da relativização das extensões a uma atribuição. Essa relativização era herdada por constituintes maiores que dominavam o pronome, podendo chegar à oração principal, como ilustrado abaixo:

(1) Ela_1 ama Pedro.

$[\![\text{ama}]\!]^g = \lambda x.\lambda y.\ \textsc{ama}(y,x)$
$[\![\text{Pedro}]\!]^g = pedro$
$[\![\text{ama Pedro}]\!]^g = [\![\text{ama}]\!]^g([\![\text{Pedro}]\!]^g)$
$[\![\text{ama Pedro}]\!]^g = \lambda y.\ \textsc{ama}(y,pedro)$
$[\![\text{ela}_1]\!]^g = g(1)$
$[\![\text{ela}_1 \text{ ama Pedro}]\!]^g = [\![\text{ama Pedro}]\!]^g([\![\text{ela}_1]\!]^g)$
$[\![\text{ela}_1 \text{ ama Pedro}]\!]^g = 1$ *sse* $\textsc{ama}(g(1),pedro)$

Nesse caso, o pronome ela_1 está livre e a sentença será pareada com condições de verdade dependentes da atribuição. Especificamente, apenas no caso de atribuições que tiverem o índice 1 em seu domínio, conferiremos um valor de verdade à sentença.

Já no capítulo 5, discutimos a interpretação composicional das orações relativas. Analisamos casos em que um pronome (resumptivo) era ligado por um índice numérico associado a um pronome relativo. Nesse caso, a interpretação

da oração relativa como um todo não dependia mais de uma atribuição. Isso era consequência do princípio composicional que chamamos de abstração funcional, como ilustrado resumidamente abaixo:

(2) $[_{\text{NP}}$ cachorro $[\underline{\text{que 1 Pedro cuida dele}}_{1}]]$
$[\![1 \text{ Pedro cuida dele}_1]\!]^g = \lambda x.\ [\![\text{Pedro cuida dele}_1]\!]^{g[1\to x]}$
$[\![\text{Pedro cuida dele}_1]\!]^{g[1\to x]} = 1$ *sse* CUIDA(*pedro*,$g[1 \to x](1)$)
$[\![1 \text{ o Pedro cuida dele}_1]\!]^g = \lambda x.$ CUIDA(*pedro*,x)

No restante desse capítulo, vamos olhar um pouco mais de perto para a ligação e sua relação com a correferência pronominal.

7.2 Correferência vs. ligação

Consideremos o par de sentenças abaixo:

(3) O João adora o chefe dele.

(4) Só o João adora o chefe dele.

Fixemos nossa atenção em casos em que, intuitivamente, o pronome se refere a João, o mesmo referente do nome próprio que aparece na posição de sujeito da sentença. O fato a se notar é que, mesmo quando restringimos nossa atenção a esse tipo de interpretação, (4) é ambígua de uma forma que (3) não é. Imagine, por exemplo, que Pedro seja o chefe do João. Nesse caso, (3) é semanticamente equivalente a (5) abaixo:

(5) O João adora o Pedro.

Já (4) admite duas interpretações nesse contexto:

(6) O João adora o Pedro e ninguém mais adora o Pedro.

(7) O João adora o Pedro e ninguém mais adora o próprio chefe.

A primeira interpretação parece a mais saliente, mas basta trabalharmos um pouquinho o contexto para que a segunda emerja naturalmente, como, por exemplo, em (8):

(8) Na firma em que o João trabalha, todos os funcionários se dão bem com os respectivos chefes. Mas, **só o João ADORA o chefe dele**.

Vejamos como abordar esses fatos com o sistema interpretativo que temos à nossa disposição. Comecemos notando que já estaríamos muito próximos do que almejamos se conseguirmos atribuir ao predicado verbal *adora o chefe dele* as duas interpretações em (9a) e (9b) abaixo, ambas funções de tipo $\langle e, t \rangle$:

(9) ⟦adora o chefe dele⟧ =
   a. λx. ADORA(*x,o chefe do joão*)
   b. λx. ADORA(*x,o chefe de x*)

Em (9a), dizemos que o pronome foi interpretado anaforicamente, sendo correferente ao sujeito João. Já em (9b), o pronome foi interpretado como uma *variável ligada*. Note que, nessa representação, tanto a posição sintática correspondente ao pronome possessivo quanto a posição do sujeito do verbo *adorar* estão preenchidas com uma mesma variável *x*, ambas ligadas pelo mesmo operador lambda (λx).

Se, agora, aplicarmos as funções em (9a) e (9b) à extensão do nome próprio *João*, ou seja, ao indivíduo João, notaremos que ambas levarão ao mesmo resultado, como mostrado a seguir:

(10) (λx. ADORA(*x,o chefe do joão*))(⟦João⟧) = 1 *sse* ADORA(*joão,o chefe do joão*)

(11) (λx. ADORA(*x,o chefe de x*))(⟦João⟧) = 1 *sse* ADORA(*joão,o chefe do joão*)

Dessa forma, quando o sujeito da oração em questão for um nome próprio, as interpretações anafórica e de variável ligada não levarão a diferenças de significado. Daí a não ambiguidade de (3) que apontamos mais acima. Já o caso de (4) é diferente, por não envolver um sujeito referencial. Para entender melhor o que está acontecendo, olhemos um pouco mais de perto para a interpretação do sintagma *só o João* em uma sentença mais simples:

(12) Só o João trabalha.

Intuitivamente, essa sentença é verdadeira se, e somente se, o João trabalha e ninguém mais trabalha. Ainda intuitivamente, podemos pensar na contribuição de *só o João* para a interpretação de (12) como uma função que inspeciona a extensão de VP e verifica se João é o único indivíduo que é levado no valor 1 por esta função. Se esse for o caso, a sentença será verdadeira. Caso contrário, será falsa. Formalizando:

(13) $[\![$Só o João$]\!]$ = $\lambda F.\ F'(joão)\ \&\ \neg\exists x[x \neq joão\ \&\ F'(x)]$
$[\![$Só o João trabalha$]\!]$ = 1 *sse*
TRABALHA(*joão*) & $\neg\exists x[x \neq joão$ & TRABALHA(x)$]$

Não vamos aqui explicitar a extensão de *só*. Como já mencionamos no capítulo 7, *só* é sintaticamente bastante flexível, podendo combinar com uma grande variedade de constituintes. Em função disso e de sua interação com aspectos prosódicos que não estamos levando em conta neste livro, sua semântica não é trivial. De qualquer forma, para os propósitos desta exposição, cujo interesse maior é na ligação pronominal, o que temos em (13) nos basta.

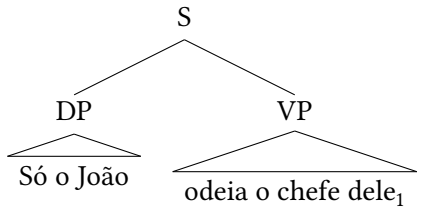
Apliquemos, então, a extensão em (13), de tipo $\langle et, t\rangle$, às duas funções (a) e (b) de tipo $\langle e, t\rangle$ em (9):

(14) $[\![$só o João$]\!](\lambda x.$ ADORA(*x,o chefe do joão*)) = 1 *sse*
ADORA(*joão,o chefe do joão*)& $\neg\exists x[x \neq joão$ & ADORA(*x,o chefe do joão*)$]$

(15) $[\![$só o João$]\!](\lambda x.$ ADORA(*x,o chefe de x*)) = 1 *sse*
ADORA(*joão,o chefe do joão*)& $\neg\exists x[x \neq joão$ & ADORA(*x,o chefe de x*)$]$

Como se pode ver, essas são exatamente as duas interpretações que havíamos detectado inicialmente, e nossa explicação para a ambiguidade de (4) e seu contraste com a não ambiguidade de (3) está praticamente pronta. Resta-nos apenas explicitar as representações sintáticas do VP *adora o chefe do João* que conduzam às interpretações em (9).

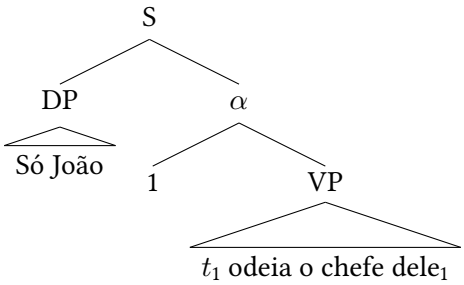
Para a interpretação de correferência, nada relevante precisa ser acrescentado. Aplicação funcional, envolvendo as extensões de VP e do DP sujeito, basta:



Nesse caso, o pronome está livre e basta assumirmos que a atribuição correspondente ao contexto relevante é tal que $g(1) = [\![$João$]\!]$.

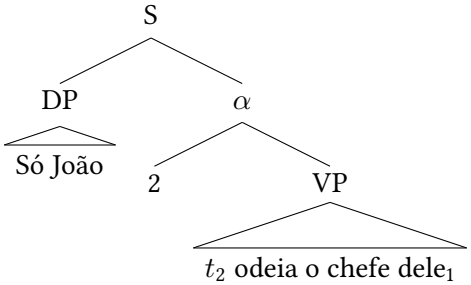
Já para a interpretação de variável ligada, precisamos acionar abstração funcional, o que, por sua vez, implica a presença de um índice derivado de movimento sintático. Aqui podemos nos valer tanto da hipótese do sujeito interno a VP que se move abertamente para sua posição superficial quanto da regra de alçamento

de quantificador, em que DPs se movem cobertamente para uma posição periférica na sentença, ambos vistos no capítulo anterior:



Nesse caso, o pronome e o vestígio estão coindexados e ambos atrelados ao mesmo índice resultante do movimento sintático. Abstração funcional se aplica nesse nível, e o constituinte α acima terá justamente a extensão que desejamos, ou seja, a interpretação vista em (9b).

Note, por fim, que a interpretação de correferência é compatível com o movimento sintático do sujeito, bastando que o pronome e o vestígio não estejam coindexados:



Fica a cargo do leitor calcular as condições de verdade dessa estrutura e conferir que são as mesmas obtidas na interpretação da estrutura sem movimento vista mais acima.

7.3 Ligação sem índices nem atribuições

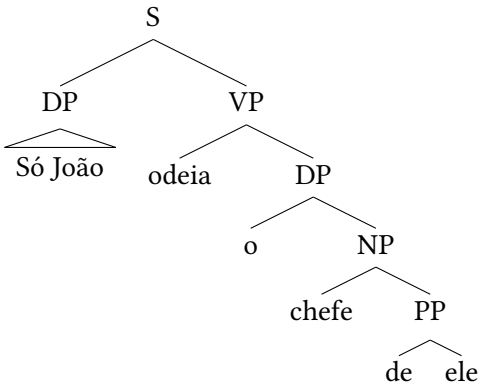
Dado tudo o que vimos até aqui, pode parecer que, para lidar com a semântica dos pronomes em geral (e dos pronomes ligados em particular), índices, atribuições e uma regra nos moldes de abstração funcional sejam essenciais para captar a interpretação desses elementos. Surpreendentemente, isso não é verdade e, conforme

mostraremos nesta seção, é possível formalizar a interpretação pronominal sem esse aparato. O que veremos a seguir é uma adaptação baseada em trabalhos de Pauline Jacobson, denominado SEMÂNTICA LIVRE DE VARIÁVEIS (*variable-free semantics*, em inglês).

Para apresentarmos essa proposta semântica, voltaremos a uma das sentenças que discutimos nas seções anteriores:

(16) Só o João odeia o chefe dele.

Sua estrutura sintática é a seguinte:



Note, desde já, a ausência de índices e vestígios de movimento sintático. Comecemos pela interpretação dos pronomes (vamos continuar ignorando traços de gênero, número e pessoa). Nesse novo sistema, um pronome denota uma **função identidade**, que mapeia cada elemento de seu domínio no próprio elemento. No caso dos pronomes pessoais singulares que estamos discutindo, trata-se de uma função de tipo $\langle e, e \rangle$ que mapeia um indivíduo x qualquer nesse próprio indivíduo x:

(17) $[\![\text{ele}]\!] = \lambda x_e.\ x$

Pronomes, portanto, não são mais expressões referenciais que denotam indivíduos. Não se preocupe se isso parece contraintuitivo. Devemos julgar uma ideia não pelas impressões iniciais, mas pelos seus resultados. Prossigamos, então, para ver aonde chegaremos. Assumindo que a preposição *de* em (16) seja semanticamente vácua, o PP *dele* herdará essa extensão:

$[\![\text{PP}]\!] = [\![\text{ele}]\!] = \lambda x.\ x$

Vejamos, agora, como essa extensão se combina com a extensão do substantivo comum *chefe*, do qual é complemento:

$[\![\text{chefe}]\!] = \lambda x.\ \lambda y.\ \textsc{chefe}(y,x)$

Essa é uma função de tipo $\langle e, et\rangle$. Como a extensão do PP é de tipo $\langle e, e\rangle$, não podemos usar aplicação funcional. Em seu lugar, vamos usar um novo princípio, que chamaremos de COMPOSIÇÃO FUNCIONAL. Antes de defini-lo formalmente e aplicá-lo a nosso exemplo, vamos entender seu funcionamento informalmente por meio de um exemplo envolvendo números.

Considere, a título de ilustração, duas funções F e G que mapeiam números naturais em números naturais, definidas da seguinte forma:

$F:\ \lambda n.\ n^2$
$G:\ \lambda n.\ 3n$

A função F mapeia um número no seu quadrado, enquanto G mapeia um número no seu triplo. Pense, agora, em uma outra função H, definida a partir de F e G da seguinte forma:

$H:\ \lambda n.\ G(F(n))$

Trata-se também de uma função de números em números. Veja que o input de H passa primeiro pela função F e que o output de F serve de input para G. O resultado final $G(F(x))$ é o output da função H. Como exemplo, tomemos $n = 2$. Teremos então o seguinte:

$H(2) = G(F(2)) = G(2^2) = G(4) = 3 \cdot 4 = 12$

Diz-se de uma função como H que ela é a composição de F e G. Note que, para que a composição funcione, é preciso que o output de uma das funções originais — F, no caso — pertença ao domínio da outra função (G). No caso acima, isso não foi problema pois ambas as funções eram funções de números para números. Note, por fim, que a ordem de composição é importante. No nosso exemplo, $G(F(x))$ e $F(G(x))$ são diferentes, como ilustrado abaixo:

$F(G(2)) = F(3 \cdot 2) = F(6) = 6^2 = 36$

Com essas preliminares no lugar, voltemos ao nosso exemplo (16) e sua derivação semântica. No ponto em que paramos, precisávamos combinar a extensão de N (*chefe*), que é de tipo $\langle e, et\rangle$ com a extensão de PP (*dele*), que é de tipo $\langle e, e\rangle$. Já sabemos que aplicação funcional não nos serve aqui. Mas note que composição funcional pode ser usada nesse caso para obtermos a extensão de NP (*chefe dele*). O output de ⟦PP⟧ é de tipo e, justamente o tipo do input de ⟦N⟧. Vejamos o resultado:

⟦NP⟧ = λx_e. ⟦N⟧(⟦PP⟧(x))

Efetuando-se as devidas substituições e conversões lambda, teremos o seguinte:

⟦NP⟧ = λx. λy. CHEFE(y,x)

Note que essa é a mesma extensão de N! De fato, compor com a função identidade é semanticamente vácuo. Se você estiver um pouco confuso, consulte nosso exemplo anterior envolvendo funções numéricas e assuma que F seja a função identidade. Como tal, $F(n) = n$ e $G(F(n)) = G(n)$. Sendo H a composição de F e G, $H(n) = G(n)$ para qualquer n. Em outras palavras, H e G são idênticas.

Já podemos agora formalizar nosso novo princípio composicional (que ainda sofrerá uma pequena modificação):

Composição funcional (versão preliminar)
Seja α um nó ramificado, cujos constituintes imediatos são β e γ. Se ⟦β⟧ é uma função de tipo $\langle a, b\rangle$ e ⟦γ⟧ é uma função de tipo $\langle b, c\rangle$, então ⟦α⟧ é uma função de tipo $\langle a, c\rangle$ e ⟦α⟧ = λw_a.⟦γ⟧(⟦β⟧(w)).

O real papel da composição funcional na interpretação pronominal fica mais claro quando lidamos com constituintes que contém pronomes, como o NP *chefe dele*, que é justamente o ponto em que estamos. Esse NP, cuja extensão vista acima é de tipo $\langle e, et\rangle$ precisa combinar com o artigo definido *o*, cuja extensão é de tipo $\langle et, e\rangle$, para obtermos a extensão do DP resultante. Novamente, aplicação funcional não pode ser usada, mas composição funcional sim. Vejamos:

⟦DP⟧ = λx_e. ⟦o⟧(⟦chefe dele⟧(x))

$[\![\text{o}]\!] = \lambda F.\ \iota y[F'(y)]$
$[\![\text{chefe dele}]\!](x) = \lambda y.\ \textsc{chefe}(y,x)$
$[\![\text{DP}]\!] = \lambda x_e.\ \iota y[\textsc{chefe}(y,x)]$

Ou, de maneira mais direta:

$[\![\text{DP}]\!] = \lambda x_e.$ *o chefe de x*

Olhando para esse resultado, fica claro que composição funcional está "passando adiante" a posição argumental ocupada pelo pronome. Note que a extensão do DP *o chefe dele* é de tipo $\langle e, e\rangle$ e que se aplicássemos essa função a um indivíduo i qualquer, o resultado seria o chefe desse indivíduo i.

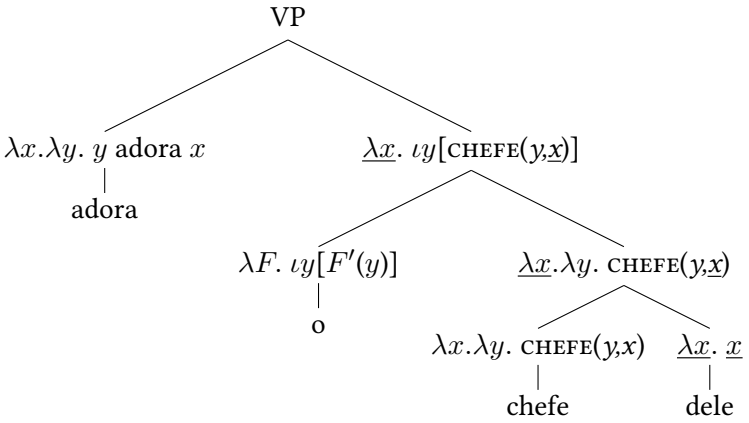
No próximo passo interpretativo, a extensão do DP objeto irá combinar com a extensão do verbo transitivo *adorar*, que representamos abaixo:

$[\![\text{adorar}]\!] = \lambda x.\lambda y.\ \textsc{adora}(y,x)$

É nesse momento que a ligação do pronome irá ocorrer semanticamente. O resultado que buscamos para o VP resultante é o seguinte:

$[\![\text{adorar o chefe dele}]\!] = \lambda x.\ \textsc{adora}(x,$*o chefe de x*$)$

Note por essa representação que a posição de sujeito (argumento externo) do verbo e a posição argumental original do pronome estão preenchidas pela mesma variável x e ligadas pelo mesmo operador lambda. É isso, efetivamente, que corresponde ao processo de ligação do ponto de vista semântico. No sistema dos capítulos e seções anteriores, isso era feito via abstração funcional com o apoio de índices e atribuições. Na ausência de tudo isso, um novo princípio precisa ser formulado que tenha o efeito de identificar essas posições argumentais que acabamos de mencionar. E aqui fica mais claro o papel que a composição funcional desempenhou. Ao passar adiante o argumento correspondente à posição original do pronome, ela permitiu que o mesmo chegasse até o constituinte (DP) que é irmão do verbo na estrutura sintática:



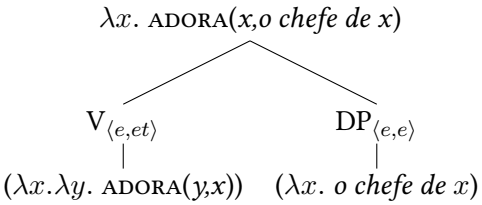
Como a extensão do verbo carrega a outra posição argumental em questão, podemos tirar partido disso e manipular ambas na formulação do princípio que chamaremos de LIGAÇÃO FUNCIONAL , definido preliminarmente abaixo:

Ligação funcional (versão preliminar)
Seja α um nó ramificado, cujos constituintes imediatos são β e γ. Se $[\![\beta]\!]$ é uma função de tipo $\langle a, et\rangle$ e $[\![\gamma]\!]$ é uma função de tipo $\langle e, a\rangle$, então $[\![\alpha]\!]$ é uma função de tipo $\langle e, t\rangle$ e $[\![\alpha]\!] = \lambda x_e.\ [\![\beta]\!]([\![\gamma]\!](x))(x)$.

Aplicando esse princípio composicional na computação da extensão do VP acima, teremos:

$[\![\text{adora o chefe dele}]\!] = \lambda x_e.\ [\![\text{adora}]\!]([\![\text{o chefe dele}]\!](x))(x)$.

Olhando para o que aparece nessa expressão após o prefixo $\lambda x.$, percebe-se que a extensão do verbo será saturada pelos indivíduos *o chefe de x* e *x*. O resultado é exatamente o que queríamos:



Como se pode notar, trata-se de uma extensão de tipo $\langle e, t\rangle$. Como passo final, para derivarmos as condições de verdade da sentença, basta combinarmos essa extensão do VP *adora o chefe dele* com a extensão do DP sujeito *só o João*, utilizando aplicação funcional, exatamente como já fizemos na seção anterior. O resultado é a leitura de variável ligada de acordo com a qual João, e ninguém mais, tem a propriedade de adorar o próprio chefe.

7.3.1 Limitando composição e ligação funcional

Conforme acabamos de ver, o trabalho conjunto de ligação funcional e composição funcional (além, claro, de aplicação funcional) permitiu derivar a leitura de variável ligada para um pronome sem o uso de atribuições nem índices. Entretanto, na forma como está, o sistema resultante gera interpretações que não são atestadas, inclusive em casos que não envolvem pronomes. Considere, por exemplo, a sentença abaixo:

(18) João odeia todo mundo.

Trata-se de uma oração simples com um verbo transitivo, um nome próprio na posição de sujeito e um DP quantificador na posição de objeto. Já vimos no capítulo anterior como interpretá-la, seja via alçamento de quantificador seja via mudança de tipos. Mas, agora, com a composição funcional no nosso leque de regras composicionais, uma derivação alternativa emerge. Acompanhe, passo a passo, a derivação a seguir:

$[\![\text{odeia}]\!] = \lambda x.\lambda y.\ \textsc{odeia}(y,x)$	tipo $\langle e, et\rangle$
$[\![\text{todo mundo}]\!] = \lambda F.\ \forall z[F'(z)]$	tipo $\langle et, t\rangle$
$[\![\text{odeia todo mundo}]\!] = \lambda x.\ [\![\text{todo mundo}]\!]([\![\text{odeia}]\!](x))$	(CF)
$[\![\text{odeia todo mundo}]\!] = \lambda x.\ \forall z[z\ \textsc{odeia}(z,x)]$	
$[\![\text{João odeia todo mundo}]\!] = [\![\text{odeia todo mundo}]\!]([\![\text{João}]\!])$	(AF)
$[\![\text{João odeia todo mundo}]\!] = 1$ *sse* $\forall z[\textsc{odeia}(z,\textit{joão})]$	

Inverteram-se os papeis argumentais! O sujeito terminou sendo interpretado como argumento interno e o objeto direto quantificado vinculado ao argumento externo do verbo.

Note que, no exemplo anterior a esse, nosso objetivo com a composição funcional foi o de passar adiante a posição argumental de um pronome até que a posição do ligador aparecesse e a ligação funcional se aplicasse. O que precisamos agora é limitar seu uso a casos envolvendo constituintes que contêm pronomes. Fazer

isso em um sistema guiado exclusivamente por tipos semânticos (*type-driven*) como o nosso não é trivial e a solução a seguir é apenas um paliativo.

Salientemos, de partida, que nesse nosso novo sistema com composição e ligação funcional, pronomes e constituintes que os contêm denotam funções que levam indivíduos (tipo *e*) em algum tipo de entidade (tipo α). Para identificar esses constituintes, não associaremos tais denotações ao tipo $\langle e, \alpha \rangle$ como fizemos até aqui, mas sim a um novo tipo $\langle e; \alpha \rangle$. Note a troca da vírgula por um ponto e vírgula. Acabamos de criar um novo construtor de tipos semânticos complexos (funcionais). Pronomes, por exemplo, serão de tipo $\langle e; e \rangle$. Podemos, assim, redefinir a composição funcional:

Composição funcional
Seja α um nó ramificado, cujos constituintes imediatos são β e γ. Se $[\![\beta]\!]$ é uma função de tipo $\langle e; a \rangle$ e $[\![\gamma]\!]$ é uma função de tipo $\langle a, b \rangle$, então $[\![\alpha]\!]$ é uma função de tipo $\langle e; b \rangle$ e $[\![\alpha]\!] = \lambda x_e.[\![\gamma]\!]([\![\beta]\!](x))$.

Composição funcional passa, agora, a ser utilizável só em casos que envolvem constituintes que contenham pronomes. De imediato, concluímos que ela não será acionada no caso de sentenças como (18) acima, que não contêm pronomes.

Também a ligação funcional pode ser limitada a casos com pronomes:

Ligação funcional
Seja α um nó ramificado, cujos constituintes imediatos são β e γ. Se $[\![\beta]\!]$ é uma função de tipo $\langle a, et \rangle$ e $[\![\gamma]\!]$ é uma função de tipo $\langle e; a \rangle$, então $[\![\alpha]\!]$ é uma função de tipo $\langle e, t \rangle$ e $[\![\alpha]\!] = \lambda x_e.\ [\![\beta]\!]([\![\gamma]\!](x))(x)$.

Note que, após a ligação, a extensão do constituinte resultante é de tipo $\langle e, t \rangle$ e não de tipo $\langle e; t \rangle$, indicando que o pronome em questão não está mais disponível para ser ligado ou passado adiante.

7.3.2 Pronomes livres novamente

Em nosso sistema anterior, com atribuições e índices, a extensão de sentenças com pronomes livres, como (19) abaixo, correspondia a um valor de verdade e era relativizada a uma atribuição.

(19) João gosta dela.

Agora, que não dispomos mais de atribuições, algo diferente acontece. Vejamos passo a passo:

$[\![\text{gosta}]\!] = \lambda x.\lambda y.\ \textsc{gosta}(y,x)$ tipo $\langle e, et\rangle$
$[\![\text{dela}]\!] = \lambda x.\ x$ tipo $\langle e; e\rangle$
$[\![\text{gosta dela}]\!] = \lambda x.\ [\![\text{gosta}]\!]([\![\text{dela}]\!](x))$ (CF)
$[\![\text{gosta dela}]\!] = \lambda x.\lambda y.\ \textsc{gosta}(y,x)$ tipo $\langle e; et\rangle$

Compare os tipos das denotações do verbo *gosta* e do VP *gosta dela*. São diferentes, já que apenas o segundo contém um pronome. Note, agora, que, como o tipo do VP é $\langle e; et\rangle$, e não $\langle e, et\rangle$, aplicação funcional não pode ser usada para combinar VP com o sujeito *João*, de tipo *e*. Nesse caso, nem mesmo composição funcional pode ser usada, já que um dos constituintes não denota uma função! Mas há uma solução. Lembre-se de que, no capítulo anterior, vimos que nomes próprios podem ser tratados como quantificadores generalizados (tipo $\langle et, t\rangle$). Para o sujeito de nossa sentença, teríamos:

$[\![\text{João}]\!] = \lambda F_{\langle e,t\rangle}.\ F'(\textit{joão})$

Sendo essa uma extensão de tipo $\langle et, t\rangle$, ela pode se combinar com a extensão do VP *gosta dela*, de tipo $\langle e; et\rangle$, via composição funcional:

$[\![\text{João gosta dela}]\!] = \lambda x.\ [\![\text{João}]\!]([\![\text{gosta dela}]\!](\text{x}))$ (CF)

$[\![\text{João gosta dela}]\!] = \lambda x.\ \textsc{gosta}(\textit{joão},x)$ tipo $\langle e; t\rangle$

Note que a sentença não tem por extensão um valor de verdade, mas sim uma função de indivíduos em valores de verdade. A ideia é que, quando uma tal sentença for enunciada em um contexto em que um indivíduo esteja saliente, obtemos um valor de verdade aplicando a função a esse indivíduo.

Terminamos, assim, nossa breve exposição simplificada de uma semântica que lida com referência e ligação pronominal sem atribuições nem índices. Note que limitamos nossa atenção a casos com apenas um pronome ligado ou livre e que não discutimos casos com mais de um pronome, sejam eles todos ligados, todos livres, ou alguns livres e outros ligados. Para tanto, seria necessário generalizar os princípios de composição e ligação funcional que apresentamos acima. Não se trata de algo conceitualmente complexo, mas que deixaremos a cargo do leitor interessado verificar através das sugestões de leitura que apresentamos a seguir.

Note, por fim, que, por um lado, esse novo sistema dispensa o uso de atribuições e índices, além da regra de abstração funcional. Por outro lado, necessita de duas novas regras, composição e ligação funcional. Podemos declarar um vencedor conceitual? Difícil dizer.

Sugestões de leitura

Discussões acessíveis sobre referência e ligação pronominal podem ser encontradas em Heim & Kratzer (1998) capítulos 9-11 e Büring (2005). A melhor introdução à semântica sem variáveis é Jacobson (2014), capítulo 15. O leitor interessado deve olhar os vários artigos de Pauline Jacobson (a principal defensora dessa abordagem) relacionados ao tema. Recomenda-se, em particular, Jacobson (1999).

Exercícios

I. Considere a sentença abaixo:

(1) Toda criança sente saudade da mãe dela.

Essa sentença admite uma leitura em que o pronome é interpretado como uma variável ligada. Derive passo a passo essa leitura tanto no sistema com índices e atribuições quanto no sistema livre de variáveis. [Para efeitos deste exercício, considere *sente saudade de* como sendo um único item lexical, semelhante a um verbo transitivo]

II. Tendo o exercício anterior em mente, considere agora a seguinte sentença:

(2) A mãe dela sente saudade de toda criança.

Diferentemente de (1), (2) não admite para o pronome *ela* uma leitura de variável ligada. A questão é se os sistemas que vimos nesse capítulo geram ou não tal leitura. Responda essa questão para os sistemas com e sem índices e atribuições. Para tanto, você precisará revisar o que vimos no capítulo anterior sobre DPs quantificadores na posição de objeto e as solu-

ções baseadas em movimento e mudança de tipos.

III. Considere as duas sentenças abaixo:

(3) Pedro se adora.
(4) Pedro adora ele.

No exercício IV do capítulo 2 foi proposta uma entrada lexical para o pronome *se* que captava seu caráter inerentemente reflexivo. Após rever esse exercício, mostre que a mesma leitura reflexiva pode ser atribuída a (4) pelo sistema sem variáveis que vimos neste capítulo. Por fim, diga se, em seu dialeto, (4) admite, ao lado de (3), essa interpretação.

Referências Bibliográficas

Abbott, Barbara. 2010. *Reference*. Oxford: Oxford University Press.

Austin, J.L. 1962. *How To Do Things with Words*. Oxford: Clarendon Press.

Barwise, John & Robin Cooper. 1981. Generalized Quantifiers and Natural Language. *Linguistics and Philosophy* 4. 159–219.

Beghelli, Filippo & Tim Stowell. 1997. Distributivity an Negation: the Syntax of *Each* and *Every*. Em Anna Szabolcsi (ed.), *Ways of Scope Taking*, 71–107. Dordrecht: Kluwer.

Büring, Daniel. 2005. *Binding Theory*. Cambridge: Cambridge University Press.

Cann, Ronnie. 1993. *Formal Semantics*. Cambridge: Cambridge University Press.

Carnie, Andrew. 2013. *Syntax: a Generative Introduction (third edition)*. Oxford: Wiley-Blackwell.

Carpenter, Bob. 1997. *Type-Logical Semantics*. Cambridge, MA: MIT Press.

Chierchia, Gennaro. 2003. *Semântica*. Campinas: Editora da Unicamp.

Chierchia, Gennaro & Sally McConnell-Ginet. 2000. *Meaning and Grammar*. Cambridge, MA: MIT Press.

Davidson, Donald. 1967. The Logical Form of Action Sentences. Em N. Rescher (ed.), *The Logic of Decision and Action*, 81–120. Pittsburgh: University of Pittsburgh Press.

Elbourne, Paul. 2013. *Definite Descriptions*. Oxford: Oxford University Press.

Frege, Gottlob. 1892. Sobre o Sentido e a Referência. Em Paulo Alcoforado (ed.), *Lógica e Filosofia da Linguagem*, Segunda Edição Revista e Ampliada, 129–158. São Paulo: Edusp.

Gamut, L.T.F. 1991. *Logic, Language, and Meaning*. Chicago: The University of Chicago Press.

Gazdar, Gerald. 1980. A Cross-categorial Semantics for Coordination. *Linguistics and Philosophy* 3. 407–409.

Geurts, Bart. 2006. Take *Five*: the meaning and use of a number word. Em Svetlana Vogeleer & Liliane Tasmowski (eds.), *Non-definiteness and plurality*, 311–329. Amsterdam/Philadelphia: Benjamins.

Geurts, Bart & Rick Nouwen. 2007. At least *et al.*: The Semantics of Scalar Modifiers. *Language* 83. 533–559.

Grice, Paul. 1975. Logic and Conversation. Em P Cole & J Morgan (eds.), *Syntax and Semantics 3: Speech Acts*, 41–58. New York: Academic Press.

Grice, Paul. 1989. *Studies in the Way of Words*. Cambridge, MA: Harvard University Press.

Haegeman, Liliane. 1994. *Introduction to Government and Binding Theory*. Oxford: Blackwell.

Heim, Irene. 1991. Artikel und Definitheit (Articles and Definiteness). Em A von Stechow & D Wunderlich (eds.), *Semantics. An International Handbook of Contemporary Research*. Berlin: de Gruyter.

Heim, Irene & Angelika Kratzer. 1998. *Semantics in Generative Grammar*. Oxford: Blackwell.

Hendriks, Herman. 1993. *Studied Flexibility: Categories and Types in Syntax and Semantics*. ILLC Dissertation Series.

Horn, Laurence. 1989. *A Natural History of Negation*. C: University of Chicago Press.

Hornstein, Norbert. 1994. *Logical Form*. Oxford: Blackwell.

Jacobson, Pauline. 1999. Towards a Variable Free Semantics. *Linguistics and Philosophy* 22. 117–184.

Jacobson, Pauline. 2014. *Compositional Semantics*. Oxford: Oxford University Press

Kennedy, Christopher. 1997. *Projecting the Adjective: The Syntax and Semantics of Gradability and Comparison*. Santa Cruz: University of California tese de doutorado.

Kratzer, Angelika. 2003. The event argument and the semantics of verbs.

Krifka, Manfred. 1999. At least some Determiners aren't Determiners. Em K. Turner (ed.), *The Semantics/Pragmatics Interface from Different Points of View*. 257–291. Elsevier Science B.V.

Larson, Richard & Gabriel Segal. 1995. *Knowledge of Meaning*. Cambridge, Mass: MIT Press.

Levinson, Stephen. 1983. *Pragmatics*. Cambridge: Cambridge University Press.

Levinson, Stephen. 2007. *Pragmática*. São Paulo: Martins Fontes.

Ludlow, Peter. 1998. *Readings in the Philosophy of Language*. Cambridge, MA: MIT Press.

Martinich, A.P. 2001. *The Philosophy of Language*. Oxford: Oxford University Press

May, Robert. 1977. *The Grammar of Quantification*. Massachusetts Institute of Technology tese de doutorado.

May, Robert. 1985. *Logical Form: Its Structure and Derivation*. Cambridge, MA: MIT Press.

Mioto, Carlos, Maria Cristina Figueiredo Silva & Ruth Lopes. 2005. *Novo Manual de Sintaxe*. Florianópolis: Editora Insular.

Mortari, Cezar. 2016. *Introdução à Lógica*. São Paulo: Editora da Unesp.

Morzycki, Marcin. 2015. *Modification*. Cambridge: Cambrigde University Press.

Munn, Alan. 1993. *Topics in the Syntax and Semantics of Coordinate Structures*. University of Maryland at College Park tese de doutorado.

Murphy, Lynne. 2010. *Lexical Meaning*. Cambridge: Cambridge University Press.

Neale, Stephen. 1990. *Descriptions*. Cambridge, MA: MIT Press.

Ostertag, Gary (ed.). 1998. *Definite Descriptions: A Reader*. Cambridge, MA: MIT Press.

Parsons, Terence. 1990. *Events in the semantics of English: a Study in Subatomic Semantics*. Cambridge, Mass.: MIT Press.

Partee, Barbara. 1986. Noun-phrase interpretation and type-shifting principles. Em J. Groenendijk, D. de Jong & M. Stokhof (eds.), *Studies in discourse representation theory and the theory of generalized quantifiers*, 115–144. Dordrecht: Reidel.

Partee, Barbara, Alice ter Meulen & Robert Wall. 1990. *Mathematical Methods in Linguistics*. Dordrecht: Kluwer.

Partee, Barbara & Mats Rooth. 1983. Generalized Conjunction and Type Ambiguity. Em *Meaning, Use, and Interpretation of Language*. Berlin: De Gruyter.

Peters, Stanley & Dag Westerståhl. 2008. *Quantifiers in Language and Logic*. Oxford: Oxford University Press.

Pires de Oliveira, Roberta. 2001. *Semântica Formal. Uma Breve Introdução*. Campinas: Mercado de Letras.

Pires de Oliveira, Roberta & Renato Basso. 2014. *Arquitetura da Conversação. Teoria das Implicaturas*. São Paulo: Parábola.

Sag, Ivan, Thomas Wasow & Emily Bender. 2003. *Syntactic Theory. A Formal Introduction*. Stanford: CSLI.

Schein, Barry. 1993. *Plurals and Events*. Cambridge, MA: MIT Press.

de Swart, Henriëtte. 2003. *Introduction to Natural Language Semantics*. Stanford: CSLI.

Szabolcsi, Anna. 2010. *Quantification*. Cambridge: Cambridge University Press.

Tarallo, Fernando. 1990. *A Pesquisa Sócio-Linguística*. São Paulo: Ática.

Tarski, Alfred. 1944. The Semantic Conception of Truth and the Foundations of Semantics. *Philosophy and Phenomenological Research* 4. 341–375.

Taylor, Kenneth. 1998. *Truth and Meaning*. Oxford: Blackwell.

Thomason, Richmond (ed.). 1974. *Formal Philosophy. Selected Papers of Richard Montague*. New Haven: New Haven University Press.

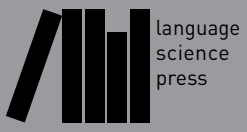
Winter, Yoad. 2001. *Flexibility Principles in Boolean Semantics*. Cambridge, MA: MIT Press.

Winter, Yoad. 2016. *Elements of Formal Semantics*. Edinburgh: Edinburgh University Press.

Did you like this book?

This book was brought to you for free

Please help us in providing free access to linguistic research worldwide. Visit http://www.langsci-press.org/donate to provide financial support or register as a community proofreader or typesetter at http://www.langsci-press.org/register.



Curso de Semântica Formal

DRAFT
of 19 de outubro de 2018, 15:59